BALANCING OF FLEXIBLE ROTORS SUPPORTED ON BEARINGS HAVING NON-LINEAR FLEXIBILITY

ARDEN E. TURPIN, B.Eng.

BALANCING OF FLEXIBLE ROTORS SUPPORTED ON BEARINGS HAVING NON-LINEAR FLEXIBILITY

By

©ARDEN E. TURPIN, B. ENG.

A thesis submitted to the School of Graduate Studies
in partial fulfilment of the
requirements for the degree of
Master of Engineering

Faculty of Engineering and Applied Sciences
Memorial University of Newfoundland
October 1991

St. John's Newfoundland Canada

ISBN 0-315-68263-9

ABSTRACT

In real-life applications, multi-disk-rotor systems are supported on bearings which have non-linear flexibility. The balancing of such systems at high speeds is a challenging task. This thesis presents a method of balancing such systems. The dynamic equations of motion of the non-linear system are derived using the influence coefficient method as well as the Lagrangian equations. An equivalent linear system is found using the optimization principles. Finally, the correct balance weights for the non-linear system are obtained based on the equivalent linear system. The results thus obtained establish the utility of such a method of balancing of non-linear systems.

ACKNOWLEDGEMENTS

I would like to thank Dr. Anand Sharan for his direction and support during my research work. I would also like to thank my wife June, for her constant encouragement and for her patient work in typing this thesis.

October 1991 Arden Turpin

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a_{ij}	elements of the matrix $[A]$
$[A]$	matrix of influence coefficients
c, C, C^{l}, C^{u}	viscous damping coefficients
C_c	critical viscous damping constant
C_i	viscous damping coefficient for the i^{th} damper
C_L	viscous damping coefficient for linearized system
$[C]$	viscous damping matrix
e_j	error at $t = t_j$
e_{jk}	error at $t = t_j$ for the k^{th} degree of freedom
F_i	force acting on the i^{th} mass
F_D	damping force
F_N	normal force
$F(t)$	forcing function
F_z, F_y	bearing forces
$\{F\}$	force vector
$\{\overline{F}\}$	complex phasor of $\{F\}$
K, K^{*}, K^{l}, K^{u}	spring stiffness coefficients
K_i	stiffness coefficient for the i^{th} degree of freedom
K_L	stiffness coefficient for linearized system
$[K_s]$	matrix of shaft stiffness coefficients

$K_{zz}^{1B},\ K_{yy}^{1B},\ K_{zz}^{2B},\ K_{yy}^{2B}$	bearing linear stiffness coefficients
$[K]$	stiffness matrix
L^*	the Lagrangian
$L,\ l_i$	shaft lengths
m_i	i^{th} disk mass
$\{m_c\}$	vector of correction masses
M	mass
$[M]$	mass matrix
n	number of rotor disks
N	number of data points
q_i	generalized displacement co-ordinate for the i^{th} degree of freedom
$\{q\}$	vector of generalized coordinates
$\{q\}_0$	vector of the value of the generalized coordinates at $t = 0$
$\{Q\}$	harmonic response vector
$\{\bar{Q}\}$	complex phasor of $\{Q\}$
r_i	eccentricity of the i^{th} rotor disk
t	time
T	kinetic energy
T_j	trial mass in the j^{th} balancing plane
$U,\ U^{**}$	objective functions

V	potential energy
V_B	potential energy of bearings
V_S	potential energy of shaft
V_T	potential energy of rotor system
$x, \dot{x}, x(t)$	displacement at time = t
$x_0, x(0)$	displacement at $t = 0$
$x_j(t)$	value of x at $t = t_j$
$x_{jk}(t)$	value of x at $t = t_j$ for the k^{th} degree of freedom
$x_L(t)$	displacement of linearized system at time = t
z_i, y_i	deflections of the i^{th} rotor disk
z_i^*, y_i^*	deflections of the i^{th} rotor disk due to bearing deflections only
$[Z]$	impedance matrix
z_{ij}	i^{th} vibration reading with trial mass installed in j^{th} balancing plane
z_{io}	i^{th} vibration reading with no trial mass
$\{z\}$	vector of residual vibration amplitudes
$\{z_0\}$	response vector due to original rotor unbalance
$\{z_m\}$	vector of amplitudes with correction masses installed
α	constant
β	constant

$\epsilon, \epsilon^{1B}, \epsilon^{2B}$	constants
ζ	damping ratio
θ_z, θ_y	constant, angular displacement
λ^*	optimal step length
μ	coefficient of friction
τ	time period
ϕ_i	phase angle of the i^{th} rotor disk
ψ	Rayleigh's dissipation function
ω	angular frequency of damped vibration
ω_n	natural frequency
$\{\ \}$	vector
$[\ ]$	matrix
$(\dot{\ })$	$\frac{d(\ )}{dt}$
$(\ddot{\ })$	$\frac{d^2(\ )}{dt^2}$
$\{\nabla U\}_i$	gradient of the objective function U at the i_{th} iteration stage
Δt	time step

CHAPTER 1

INTRODUCTION AND LITERATURE SURVEY

1.1 Introduction

The major cause of rotor vibrations in rotating shafts is residual unbalance. These unbalances in the rotor can be due to material inhomogeneities, manufacturing processes, keyways, slots etc. Also, during the operation of the rotor, there is a deterioration in the balance of the rotor due to component wear, thermal bending, and process contamination, all of which cause an increase in the vibratory response of a rotor. In addition to causing vibration in the rotor shafts, the unbalance forces are transmitted to the rotor bearings and support structure. The purpose of rotor balancing is to reduce these unbalance forces and to obtain acceptable levels of vibration for reliable operation of rotating equipment.

The development of computational models to identify unbalance forces in rotating machinery without the need for numerous trial runs, can be a valuable tool in the successful balancing of rotors in the field. A successful model will enable the unbalance to be estimated from the displacement data at the balancing speed, thus balancing can be carried out without the need for numerous trial runs to estimate the influence coefficient matrix.

The rotor model used for the rotor response computations usually incorporates linear bearing support characteristics. However many rotors experience non-linear forces at the rotor interactive devices such as bearings and seals. For example, rolling element bearings are commonly used in the various kinds of turbomachinery such as compressors and gas turbine engines. The spring characteristics of the rolling element bearing is non-linear and the vibration response of the rotor has to be calculated taking this non-linearity into account. The non-linear problem will have to be solved numerically as a function of time, to obtain the steady state response. In addition, the current balancing techniques of modal balancing and influence coefficient balancing are based on the assumption of linear superposition of rotor response. If the rotor non-linearities are considered, it would be a difficult to balance the rotor since linear superposition would not be valid.

1.2 Literature Survey

1.2.1 Modal Balancing

Flexible rotor balancing theory was first developed by Meldahl in 1954. He showed the orthogonality conditions between any two modes, and that it was possible to balance a flexible rotor mode by mode. Beginning in 1959 this method was further developed by Bishop, and his co-workers. These papers greatly advanced the theory and application of the modal balancing method and resulted in the development of practical techniques for flexible rotor balancing.

In the first of these papers, Bishop (1959) formulated the whirling amplitudes of a Jeffcot rotor model as a modal series. Gladwell and Bishop (1959) applied this analysis to an axi-symmetric shaft of non-uniform cross-section in flexible bearings, showing how to find the natural frequencies and the modal functions. Then Bishop and Gladwell (1959) presented the general method of modal balancing. This paper analytically examines the problems with low speed balancing of flexible rotors, and examines the effects of a bent shaft and shaft weight. This theory was further developed in the following decade when Bishop and Parkinson (1972) presented a review of the modal balancing technique, in particular, the N plane method of modal balancing. Also in that year Kellenberger (1972) published a paper on the theoretical explanation and justification of the N-plane and alternately the N + 2 plane modal balancing techniques. The various procedures for modal balancing generally fall into these groups.

1.2.2 Influence Coefficient Method

The influence coefficient method of flexible rotor balancing utilizes the rotor-bearing sensitivity for the calculation of balance correction weights. At first, the unbalance response at the rotor measuring planes is determined at a given speed without any correction weights. A trial weight is applied in one of the balancing planes and the rotor response is obtained for all of the measuring planes. This process is repeated for all of the balancing planes and from this data the influence coefficient matrix is obtained. The required correction weights can be calculated by multiplying the inverse of the influence coefficient matrix with the original unbalance response vector. These coefficients, once available may be used for rebalancing that rotor at a future time, or all other rotors of that family. In many cases, the influence coefficients may be calculated using a proven unbalance response computer program, when accurate analytical models of the rotor system are available.

In 1964, Goodman developed an influence coefficient balancing procedure which used a least squares method to minimize the residual rotor amplitudes at selected locations. Rieger (1967) developed an efficient calculation procedure to implement the influence coefficient method. Rieger also made an analytical study of the effectiveness of the influence coefficient method. In 1972 Lund and Tonneson and in 1974 Tonneson, presented the results of experiments which examined the accuracy and validity of the influence coefficient method. Tessarzik et al. (1972) reported on a test program which studied the exact point speed influence coefficient method and Tessarzik and Badgley (1974) evaluated the least squares influence coefficient procedure. Generally, the least squares method was more accurate than the exact point method. Analytical and experimental analysis continued on the use of the influence coefficient method. One notable variation to the procedure was presented by Little and Pilkey (1976). They presented a linear programming approach for a rotor system where the number of balancing planes exceeded the number of measurement stations. In this procedure the optimum solution is chosen such that an objective function is minimized.

1.2.3 Equivalent Linearization

As an alternative to using numerical integration procedures, many linearization techniques have been developed to approximate the dynamic response of non-linear systems. Among the earliest methods proposed were (a) the method of Krylov and Bogoliubov (1943), (b) the perturbation methods (Jones, 1978), (c) the Duffing method (Stoker, 1950), (d) methods using ultraspherical polynomials (Sinha and Srinivasan, 1971-a), and (e) the weighted mean square method (Sinha and Srinivasan, 1971-b). The text by Nayfeh and Mook (1979) is a good reference on these early methods.

Bandstra (1983) compared the use of equivalent viscous damping for non-linear damping in discrete and continuous vibrating systems. The equivalent viscous damping coefficient was found such that the equivalent system would dissipate the same amount of energy per cycle as the nonlinear system. Rakheja et al. (1985) presented a discrete harmonic linearization technique, which is an iterative technique to establish equivalent local viscous damping coefficients as functions of frequency, amplitude of excitation, and the type of nonlinearity. Rao (1985) used the concept of strain energy correlation to find an equivalent linear term for a non-linear spring of a single degree of freedom system. Géradin and Kill (1988) described the application of the harmonic balance method for the linearization of common nonlinear rotor and stator interactive forces. Guang et al. (1988) demonstrated the use of equivalent linearization of bearing squeeze film damping forces in the solution of the synchronous response of a flexible rotor.

1.3 Objectives of This Study

The influence coefficient method and the modal balancing methods are the two principal methods of flexible rotor balancing today for linear systems, with the more practical method being the influence coefficient method. Both methods depend on important principles of linear systems, the principles of proportionality and superposition. However, the effects of a rotor's non-linearities must be considered in any rotor balancing procedure.

This thesis presents an equivalent linearization method that can be applied to non-linear multi-degree-of-freedom rotor systems. It also demonstrates how the equivalent linearization technique in conjunction with computational models of non-linear rotors can be used in balancing. The objectives of the thesis can be summarized as follows:

1. To develop an equivalent linearization method which can be easily adapted to multi-degree-of-freedom rotor dynamic problems.

2. To develop a mathematical model of a multi-mass rotor supported on non-linear roller bearings, and then find an equivalent linear model for the non-linear rotor.

3. To balance the non-linear rotor using the equivalent linear system.

In Chapter 2 an equivalent linearization method is presented. The equivalent linearization procedure is formulated as an optimization problem, where the optimization is based on the minimum error between the exact or numerical solution and the equivalent linear system. The accuracy and simplicity of this method for a variety of dynamic problems is demonstrated in this chapter, comparing it to other linearization procedures for non-linear dynamic problems.

The mathematical formulation of the nonlinear rotor system's dynamic equations of motion is presented in Chapter 3. These equations for the unbalanced damped response of the rotor disks are developed using the influence coefficient method (Sharan & Rao, 1985). Also in Chapter 3, the theory of balancing linear rotors using the least squares method is presented. Then a procedure for balancing a rotor supported on nonlinear bearings using the equivalent linearization procedure and the least squares balancing method is demonstrated. The conclusions and the recommendations for future work are given in Chapter 4.

CHAPTER 2

EQUIVALENT LINEARIZATION OF NON-LINEAR SYSTEMS

2.1 Introduction

The first objective of this work is to develop an equivalent linearization method which can be easily adapted to non-linear multi-degree-of-freedom rotor dynamic problems. As discussed in the literature review, there are several methods of solving non-linear systems which involve some form of linearization. The solutions discussed apply to a particular conservative or non-conservative system and to single-degree-of-freedom systems, or at most, to two-degree-of-freedom systems. In rotor dynamic problems one has to solve multi-degree-of-freedom systems; therefore there is a definite need for a method which can be used for single-degree-of-freedom or multi-degree-of-freedom systems and also for conservative and non-conservative systems, and for free and steady state vibration problems. This chapter describes such a method, and compares the results with those of other researchers. The method developed is first applied to a single-degree-of-freedom system, and then to multi-degree-of-freedom systems.

2.2 Linearization Using Optimization Techniques

2.2.1 Mathematical Modelling of A Single-Degree-of- Freedom System

Consider the free vibration of a conservative non-linear single-degree-of-freedom system (SDOF). The first task in the linearization process is to find the exact solution, if available, or else one can use numerical techniques such as Runge-Kutta method to obtain the time response $x_j(t)_R$, where $j = 1...N$, and N is the number of points selected from the time period τ. In the next step, one assumes an equivalent linear system, i.e. the stiffness and the damping parameters are assumed, and the corresponding time history $x_j(t)_{Eq}$ is calculated. Defining the error e_j as

$$e_j = x_j(t)_R - x_j(t)_{Eq} \tag{2.1}$$

one defines an objective function U as

$$U = \sum_{j=1}^{N} e_j^2 \tag{2.2}$$

For the sake of completeness, we mention here the expression for $x_j(t)_{Eq}$, which can be obtained from any standard textbook (single-degree solution) as

$$x_j(t)_{Eq} = \exp^{-\zeta\omega_n t}(X_0 \cos\omega_d t + \left[\frac{\dot{X}_0+\zeta\omega_n X_0}{\omega_d}\right]\sin\omega_d t) \tag{2.3}$$

For undamped systems, ζ will be equal to zero.

Minimization Algorithm

For the general case one can define a vector $\{V\}$ whose components are stiffness and damping, K and C respectively, expressed as

$$\{V\} = \begin{Bmatrix} K \\ C \end{Bmatrix} \tag{2.4}$$

Now one can state the minimization problem as:

Minimize the objective function

$$\begin{aligned} U &= f(K, C) \\ &= f(x_1, x_2) \end{aligned} \tag{2.5}$$

subject to the constraints

$$\begin{aligned} K^l &< K < K^u \\ C^l &< C < C^u \end{aligned} \tag{2.6}$$

Such types of problems can be solved using either direct search or gradient optimization methods. One of the gradient methods is the Davidon-Fletcher-Powell Method (Rao, 1978), which is widely known to be a dependable method and therefore has been used in this work.

A step by step procedure for achieving such a goal at the i_{th} iteration stage can be given as:

(1) Assume a vector $\{V\}$ and a (2×2) positive symmetric matrix $[H]$. If the number of elements in the vector $\{V\}$ were n, then $[H]$ would be of the size $n \times n$.

(2) Calculate the gradient of the objective function, $\{\nabla U\}$ at the point $\{V\}_i$ and evaluate $\{G\}_i$ using

$$\{G\}_i = -[H]_i\{\nabla U\}_i \tag{2.7}$$

(3) Obtain the optimal step length λ^* in the direction $\{G\}_i$ and set

$$\{V\}_{i+1} = \{V\}_i + \lambda_i^*\{G\}_i \tag{2.8}$$

(4) Compute the objective function corresponding to $\{V\}_{i+1}$ and test for optimality. If it is achieved within the required tolerance then stop, otherwise go to the next step.

(5) Update the matrix $[H]_{i+1}$ for the next iteration using the relation

$$[H]_{i+1} = [H]_i + [L]_i + [N]_i \tag{2.9}$$

where

$$[L]_i = \frac{\lambda_i^*\{G\}_i\{G\}_i^T}{\{G\}_i^T\{P\}_i} \tag{2.10}$$

$$[N]_i = \frac{-\{[H]_i\{P\}_i\}\{[H]_i\{P\}_i\}^T}{\{P\}_i^T[H]_i\{P\}_i} \tag{2.11}$$

$$\{P\}_i = \nabla U_{i+1} - \nabla U_i \tag{2.12}$$

(6) Go to step 2

The above mentioned method is quite stable and converges quadratically. As a point of clarification, it should be noted here that for conservative systems, i.e. $C = 0$, Eq. (2.5) will be reduced to a single parameter optimization only.

2.2.2 Mathematical Modelling of Multi-Degree-of-Freedom Systems

For a multi-degree-of-freedom system, one can write equations similar to Eqs. (2.1) and (2.2) as

$$e_{jk} = x_{jk}(t)_R - x_{jk}(t)_{Eq} \tag{2.13}$$

and

$$U = \sum_{k=1}^{n} \sum_{j=1}^{N} e_{jk}^2 \tag{2.14}$$

where the indices j and k stand for the number of points selected in τ and the degrees of freedom in the system, respectively. The vector $\{V\}$ in this case will be given by

$$\{V\} = \begin{Bmatrix} K_1 \\ C_1 \\ \cdot \\ K_n \\ C_n \end{Bmatrix} \tag{2.15}$$

To obtain $x_{jk}(t)_R$, one can again use the Runge-Kutta method applicable to system of equations, whereas $x_{jk}(t)_{Eq}$ is obtained from the modal analysis of the equivalent system of linear differential equations given by

$$[M]\{\ddot{x}(t)\} + [C]\{\dot{x}(t)\} + [K]\{x(t)\} = \{F(t)\} \tag{2.16}$$

Thus the steps involved in this method for the n-degrees-of-freedom system are:

(1) Assume a vector $\{V\}$, given by Eq. (2-15), within the upper and lower limits for the damping and stiffness parameters.

(2) Obtain the time history of each of the degrees-of-freedom within a time period using modal analysis of Eq. (2-16).

(3) Calculate U given by Eq. (2-14)

(4) Follow steps 1-6 in the minimization algorithm mentioned in the Section 2.2.1

For transient or forced vibration response one can always calculate U using Eqs.

(2-13) and (2-14), and of course, $x_{jk}(t)_{Eq}$ must be obtained corresponding to an appropriate forcing function.

2.2.3 Case Studies

2.2.3.1 Case No.1 - Free Vibration of a Conservative Single-Degree-of-Freedom System With Cubic Stiffness Non-Linearity

Consider a conservative, undamped free vibration problem with a cubic spring element. The equation of motion is as follows:

$$M\ddot{x} + Kx + K^{*}x^{3} = 0 \tag{2.17}$$

which can be written as

$$\ddot{x} + \left(\frac{K}{M}\right)x + \left(\frac{K^{*}}{M}\right)x^{3} = 0 \tag{2.18}$$

where $M = 0.01;\ K = 2.0;\ K^{*} = 0.5;$ with initial conditions given as

$\dot{x}(0) = 0;\ \ x(0) = 1.5.$

The system response $x(t)$ and time period for this equation is found using:

1. A Modified Perturbation Method
2. A Weighted Mean Square Method
3. The Optimization Method

Each of the methods are discussed first, and then the results can be compared to those obtained using the numerical solution (the Runge-Kutta method).

Modified Perturbation Method

In a paper by Jones (1978), a method is presented which is a variation on the perturbation expansions to find a solution to a non-linear equation. The paper discusses the periodic solution of the equation

$$\ddot{x} + \omega^2 x + \epsilon x^3 = 0 \tag{2.19}$$

with initial conditions as $x(0) = A$ and $\dot{x}(0) = 0$. The method described was devised to extend the range of validity of perturbation expansions to include larger values of ϵ. From this method by Jones, an approximate solution for Eq. (2.19) becomes

$$x \approx A\cos\tau + \left(\frac{rA}{24}\right)(\cos 3\tau - \cos\tau) \tag{2.20}$$

where $$\tau = \frac{\omega t}{\sqrt{1-r}}, \quad r = \frac{J}{(1+J)} \quad and \quad J = \frac{3A^2\epsilon}{4\omega^2}. \tag{2.20a}$$

Eq. (2.20) can be applied to Case No. 1 where the equation parameters would be as follows:

$$\begin{aligned} \epsilon &= \frac{K^*}{M} \\ \omega^2 &= \frac{K}{M} \\ x(0) &= A = 1.5 \\ \dot{x}(0) &= 0 \end{aligned}$$

Weighted Mean Square Method

In the paper by Sinha and Srinivasan (1971-b), a linearization technique is presented which yields an approximate period of oscillation τ for the problem of free oscillations of conservative systems, governed by the differential equation :

$$\left(\frac{d^2x}{dt^2}\right) + f(x) = 0 \tag{2.21}$$

with initial conditions as $\dot{x}(0) = 0$ *and* $x(0) = A$. For the case of cubic non-linearity, the nonlinear function $f(x)$ in Eq. (2.21) is given as

$$f(x) = ax + bx^3 \tag{2.22}$$

The approximate period of oscillation for the case of cubic non-linearity is found to be

$$\tau = 2\pi\left[a + \left(\frac{(m^*+3)}{(m^*+5)}\right)bA^2\right]^{-\frac{1}{2}} \tag{2.23}$$

where when $m^* = 3$ is recommended to be used for the most accurate results.

This method of approximating the period of oscillation can be applied to Case I, given by Eq. (2.18). With $m^* = 3$, the approximate time period τ for Eq. (2.18) becomes

$$\tau = 2\pi\left[a + \left(\frac{(3b)}{4}\right)A^2\right]^{-\frac{1}{2}} \tag{2.24}$$

where, $a = K/M;\; b = K^*/M;\; x(0) = A = 1.5.$

Using this approximate period of motion obtained for the interval $(A,-A)$, a linearized equation of motion can be written as

$$\ddot{x} + \omega_L^2 x = 0 \tag{2.25}$$

where $\omega_L = \tau/2\pi$. Therefore the approximate solution for Eq. (2.17) becomes

$$x(t) = A\cos\omega_L t \tag{2.26}$$

Optimization Method of Linearization (Sharan and Turpin, 1989)

An approximate linearized function for Eq. (2.17) can be written as

$$M\ddot{x} + K_L x = 0 \tag{2.27}$$

which can be written as:

$$\ddot{x} + \omega_L^2 x = 0 \tag{2.28}$$

where

$$\omega_L = \sqrt{\frac{K_L}{M}},$$

with the initial conditions $x(0)=0$ *and* $\dot{x}(0)=0$, the solution of the approximate linearized equation becomes

$$x_L(t) = A\cos\left(\sqrt{\left(\frac{K_L}{M}\right)}t\right) \tag{2.29}$$

The exact solution of the nonlinear Equation (2.17) was found using the Runge-Kutta method. A number of points (x,t) were selected from the solution for the exact system response. Using the optimization technique for a SDOF system, the linearized stiffness parameter K_L in Eq. (2.27) was found so as to minimize the error between the exact solution and the approximate solution. The value of K_L for this example was 2.83.

Table 2.1 shows the calculated time period of the free vibration of the

Table 2.1 Free Vibration of a Single-Degree-of-Freedom System - Case No. 1

Equation of Motion: $M\ddot{x} + Kx + K^{*}x^{3} = 0$, Time Period $= \tau$ (secs)

$$M = 0.01\,(Kg) \qquad \dot{x}(0) = 0$$
$$K = 2.0\,(\frac{N}{m}) \qquad x(0) = 1.5$$
$$K^{*} = 0.5\,(\frac{N}{m^{3}})$$

Parameter	Perturbation Method	Weighted Mean Square Method	Optimization Method
τ (s)	0.3726	0.3726	0.3726
U	0.0012	0.00178	0.00047

conservative single-degree-of-freedom system obtained from the three linearization techniques as well as from the optimization method. The value of the objective function. U is also shown in this Table. Here U is the square of the error per point, i.e. it is equal to U/N, where N is the number of points in a given period. The calculated period of vibration is the same for all three methods, but the error U for the optimization method is the least.

2.2.3.2 Case No.2 - Forced Vibration of a Non-Conservative Single-Degree-of-Freedom System With Cubic Stiffness Non-Linearity

The equation of motion, and the assumed parameters, for a non-conservative single-degree-of-freedom system subjected to a harmonic forcing function are given in Table 2.2. In this case study the steady amplitude for the non-linear system was found using the optimization method and the weighted mean square method (Sinha and Srinivasan, 1971-b). Table 2.2 shows that the amplitudes were the same in both cases, but both were less than the amplitude calculated using the Runge-Kutta numerical solution.

2.2.3.3 Case No.3 - Forced Vibration Response of a Single-Degree-of-Freedom System With Non-linear Damping and Cubic Stiffness Non-linearity

Consider a non-linear non-conservative system with a mixed type of non-linear damping and a cubic spring subjected to step function excitation as follows:

$$M\ddot{x} + \alpha(1+x^2)\dot{x} + K_1 x + \beta x^3 = Fu(t) \tag{2.30}$$

where $u(t) = 1$ for $t > 0$ and $u(t) = 0$ for $t < 0$

Table 2.2 Steady State Amplitude of Nonlinear Forced Vibration Problem - Case No. 2

Equation of Motion: $M\ddot{x} + c\dot{x} + ax + bx^3 = F_0 \sin\omega t$

$$M = 453.23\,(Kg) \qquad F_0 = 4448.2\ N$$

$$\omega = 37.70\left(\frac{rad}{sec}\right) \qquad C = 11390.2\left(\frac{N\text{-}sec}{m}\right)$$

$$a = 1{,}789{,}271\left(\frac{N}{m}\right) \quad b = 1.2 \times 10^{10}\left(\frac{N}{m^3}\right)$$

Parameter	Weighted Mean Square Method (mm)	Optimization Method (mm)	Runge Kutta Method (mm)
X_{ss}	3.37	3.37	3.397

$M = 1.0, F = 0.2, \alpha = 0.1$

$K_1 = 1.0, \beta = 1.0$

with initial conditions $x(0) = 0, \text{ and } \dot{x}(0) = 0$

The system response $x(t)$ for Eq. (2.30) is found using an application of ultraspherical polynomials and by the optimization method of linearization. The results are compared to the exact solution found from the Runge-Kutta method.

Ultraspherical Polynomials Method

An approximate solution of Eq. (2.30) was presented by Sinha and Srinivasan (1971-a). In the analysis the differential equations for amplitude and phase were formulated by the method of the variation of parameters. The approximate solutions are obtained by a generalized averaging technique based on ultraspherical polynomial expansions. From this paper the approximate response for Eq. (2.30) is given by

$$x = A_0 - A(t^*)\cos(1+n)\,t^* \tag{2.31}$$

where the non-dimensional time t^* is given by

$$t^* = \sqrt{\frac{K_1}{M}} * t \tag{2.32}$$

In Eq. (2.31), A_o is the real root of the equation

$$x + \beta x^3 - P = 0 \tag{2.33}$$

where

$$P = Fu(t), \tag{2.34}$$

$$A = -A_0, \text{ and} \tag{2.35}$$

$$n = \frac{3}{2}A_0^2 + \frac{3}{8}A^2 \tag{2.36}$$

The function $A(t^*)$ is given by

$$A(t^*) = \frac{A_0 e^{-\left(\frac{\alpha}{2}\right)(1+A_0^2)t^*}}{\left[1+\frac{1}{4}\left(\frac{A_0^2}{1+A_0^2}\right)(1-e^{-\alpha(1+A_0^2)t^*})\right]^{\frac{1}{2}}} \tag{2.37}$$

Optimization Method of Linearization

In the optimization method an approximate linearized function for Eq. (2.30) is written as:

$$M\ddot{x} + C_L\dot{x} + K_L x = F_0 \tag{2.38}$$

where $F_o = Fu(t) = 0.2 \text{ at } t > 0$

$\dot{x}(0) = 0$

$x(0) = 0$

The general solution to the linearized equation (Rao, 1986-a) is

$$x_L(t) = \left(\frac{F_0}{(K_L\sqrt{1-\zeta^2})}\right)\left[\sqrt{1-\zeta^2} - e^{-\zeta\omega_n t}\cos(\omega_d t - \phi)\right] \tag{2.39}$$

where,

$$\phi = \tan^{-1}\left(\frac{\zeta}{\sqrt{1-\zeta^2}}\right)$$

$$\omega_d = \sqrt{1-\zeta^2} * \omega_n$$

$$\omega_n = \frac{K_L}{M}$$

$$C_c = 2M\omega_n$$

$$\zeta = \frac{C_L}{C_c}$$

Eq. (2.30) was solved using the Runge-Kutta method and multi-parameter optimization was used to find the linearized damping constant C_L and the linearized spring constant K_L in Eq. (2.38), to minimize the error between the Runge-Kutta solution and the approximate solution $x_L(t)$.

Fig. 2.1 gives the results for $x(t)$ found from the ultraspherical polynomials method, the optimization method of linearization, and from the Runge-Kutta numerical solution. Graphically, the two linearization procedures produce similar results to the numerical solution. The error between the approximate solutions and the numerical solution, U, was calculated for a selected time period between *3* and *9* seconds. For this period the error U was 0.001031 and 0.001029 for the ultraspherical polynomials method and the optimization method respectively.

2.2.3.4 Case No.4 - Forced Vibration of A Single-Degree-of-Freedom System With Non-linear Damping

Consider the forced vibration of damped single-degree-of-freedom system as follows

$$M\ddot{x} + Kx + f(\dot{x},x,t) = F_0 \sin\omega t \qquad (2.40)$$

where $f(\dot{x},x,t)$ represents the linear and non-linear damping. Damping exists whenever there is energy dissipated, such as from external friction, fluid viscosity, and internal friction. Bandstra (1983) presented the results of a study of a system described by Eq. (2.40), with non-linear damping. In this study he compared the exact solution found from the Runge-Kutta numerical procedure, to the approximate solution using equivalent viscous damping found from energy balance methods. The following types of non-linear damping were considered in the study:

1. Coulomb Damping

Coulomb damping is used to represent dry friction present in sliding surfaces such as laminated springs and structural joints. In Coulomb damping, the force resisting the motion is assumed to proportional to the normal force between the sliding surfaces and independent of velocity except for the sign. The Coulomb damping force is represented by

$$F_D = (sign\ \dot{x})\, F_c \qquad (2.41)$$

where $F_c = \mu F_N$, and $(sign\ \dot{x})$ has the value unity and the sign of the velocity. F_N is the normal force and μ is the coefficient of friction.

2. Velocity Squared Damping

Velocity squared damping is present when a mass vibrates in a fluid or when fluid is passed rapidly through an orifice. The damping force is given as

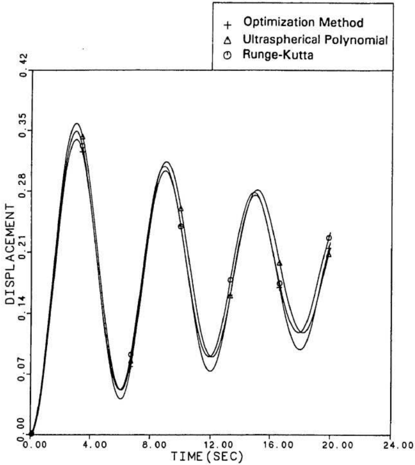


Fig. 2.1 System Response: Case No.3

$$F_D = (sign\ \dot{x})\, a\dot{x}^2 \tag{2.42}$$

Note that viscous damping is a special case of velocity to the n power damping, where the fluid flow is relatively slow, and therefore laminar. In this case the damping force is

$$F_D = c\dot{x} \tag{2.43}$$

3. <u>Solid Damping</u>

Solid damping represents the energy dissipated within a material and it is expressed as

$$F_D = (sign\ \dot{x})\, b\ |x| \tag{2.44}$$

4. <u>Displacement Squared Damping</u>

Displacement squared damping is similar to solid damping and the damping force is expressed as:

$$F_D = (sign\ \dot{x})\, dx^2 \tag{2.45}$$

Bandstra (1983) found the equivalent linear viscous damping coefficient for the four types of non-linear damping using the equivalent energy method. The energy dissipated per cycle by the non-linear damping forces is equated to the linear viscous energy dissipation; and therefore an equivalent linear viscous damping coefficient can be found, and the linearized system response calculated.

In the optimization method as before, the numerical solutions were found and the equivalent linear damping coefficient was found such that the square of the errors between the numerical solution and the linearized system's response was minimized.

Table 2.3 shows the steady state amplitude response for the various types of forced nonlinear damped systems. In column 2, the steady state amplitude response obtained by the Runge-Kutta Method is shown. The equivalent viscous damping parameter and the steady state amplitude found by using the energy balance method is shown in columns 3 and 4 respectively. The error between the Runge-Kutta solution and the results in column 4 are given in column 5. Similar results are reported in columns 6,7, and 8, for the single variable optimization method. The error was less for the optimization linearization method.

The results in column 8 were obtained by varying the damping parameter only because the stiffness term was linear in the specified problem. In fact, one could also obtain better response characteristics in the simulated model if the stiffness parameter was also considered a variable, which is represented as K; the rationale being that we want to come up with an equivalent system whose response behaviour is closest to the nonlinear system. These results are shown in the columns 9 - 12. It is especially true in the two lower cases, i.e. when the system has displacement squared or solid damping.

2.2.3.5 Case No.5 - Free Vibration of a Multi-Degree-of-Freedom System With Non-Linear Damping

To illustrate the applicability of the optimization method to a multi-degree-of-freedom system analysis, a three-degree-of-freedom example with non-linear damping was chosen as shown in Fig. 2.2. The values for the various parameters and the initial conditions

Table 2.3 Forced Vibration of Single-Degree-of-Freedom Systems with Nonlinear Damping - Case No. 4

Equations of Motion

Coulomb Damping: $m\ddot{x}+Kx+(\text{sign }\dot{x})F_c=F_0\sin\omega t$

Velocity Squared Damping: $m\ddot{x}+Kx+(\text{sign }\dot{x})a(\dot{x})^2=F_0\sin\omega t$

Disp. Squared Damping: $m\ddot{x}+Kx+(\text{sign }\dot{x})b(x)^2=F_0\sin\omega t$

Solid Damping: $m\ddot{x}+Kx+(\text{sign }\dot{x})d(x)=F_0\sin\omega t$

$m=453.23\ kg$

$K=1789271.0\frac{N}{M}$

$f_n=10\ Hz$

$F_0=4448.2\ N$

$\omega=37.70\ \frac{rad}{sec}$

X_{ss}=*Steady State amplitude (mm)*

C_{eq}=*Equivalent Viscous Damping*

K_m=*Modified Stiffness,* $(\frac{KN}{m})$

$a=94458.2$

$b=1.38\cdot10^8$

$d=4.93\cdot10^5$

$F_c=1163.2\ N$

Type of Damping	Runge Kutta Soln	Energy Balance Method			Single Variable Optimization			Dual Variable Optimization			
1	2	3	4	5	6	7	8	9	10	11	12
	X_{ss}	C_{eq}	X_{ss}	% Error	C_{eq}	X_{ss}	% Error	C_{eq}	K_m	X_{ss}	% Error
Coulomb	3.82	11390.2	3.64	4.96	10032.73	3.69	3.4	10014.14	1789.27	3.88	+ 1.6
Velocity Squared	3.72	11390.2	3.64	2.15	11074.37	3.65	1.88	11068.05	1775.95	3.69	- 0.8
Disp. Squared	4.53	11390.2	3.64	20.70	6861.17	3.79	16.30	6580.35	1548.26	4.74	+ 2.0
Solid	4.20	11390.2	3.64	14.55	8182.66	3.75	10.7	8024.2	1625.97	4.33	+ 3.0

are given in Table 2.4. Using the optimization method of linearization, $x_{jk}(t)_R$ in Eq. (2.13) was obtained using the Runge-Kutta solution while $x_{jk}(t)_{Eq}$ was obtained from the modal analysis of the equivalent system of linear differential equations similar to Eq. (2.16). See Appendix A for a description of the modal analysis procedure. The objective function U from Eq. (2.14) was divided by 3N, and a new objective function U^{**} was defined as

$$U^{**} = \frac{\sum_{k=1}^{n} \sum_{j=1}^{N} e_{jk}^{2}}{3N} \tag{2.46}$$

The reason for using *3* was due to the three variables being used. The response plots for the three variables are shown in Figs. 2.3 to 2.5, where the Runge-Kutta solutions are also shown. As one can see in these figures, the optimization technique yields good results. U for this example was 2.92×10^8 and N selected was equal to *40*.

2.3 Conclusions

In this chapter, the vibration response of non-linear single-degree or multi-degree-of-freedom systems, whether conservative or non-conservative, free and forced, were formulated as optimization problems and solved. In these formulations, the optimization was based on the minimum error between the exact or numerical solution and an equivalent linear system solution. The results were compared, wherever possible, with those obtained by other researchers. The results show that this method is very versatile and relatively more accurate than the existing linearization methods.

Table 2.4 Free Vibration of a Multi-Degree-of-Freedom System with Nonlinear Damping - Case No. 5

Parameter	Values (kg)	Parameter	Value (N/m²)	Parameter*	Values
M_1	4500	K_1	1,789,000	C_1	45,000 (N-s²/m²)
M_2	4300	K_2	1,400,000	C_2	27,000 (N-s²/m²)
M_3	1000	K_3	1,000,000	C_3	55,000 (N-s²/m²)
		K_4	2,000,000	C_4	9.0×10^6 (N-s²/m²)
				C_5	6.0×10^6 (N-s²/m²)

* C_1, C_2, C_3 are constants for velocity squared damping and C_4, and C_5 are constants for displacement squared damping

EQUATIONS OF MOTION

1. $M_1\ddot{x}_1 + (K_1 + K_2)x_1 - K_2x_2 + (C_1 + C_2)(sign\ \dot{x}_1)(\dot{x}_1)^2 + C_5(sign\ \dot{x}_1)(\dot{x}_1)^2 -$
   $C_2(sign\ \dot{x}_2)(\dot{x}_2)^2 - C_5(sign\ x_3)(x_3)^2 = 0$
2. $M_2\ddot{x}_2 - K_2x_1 + (K_2 + K_3)x_2 - K_3x_3 - C_2(sign\ \dot{x}_1)(\dot{x}_1)^2 + (C_2 + C_3)(sign\ \dot{x}_2)(\dot{x}_2)^2$
   $C_3(sign\ \dot{x}_3)(\dot{x}_3)^2 = 0$
3. $M_3\ddot{x}_3 - K_3x_2 + (K_3 + K_4)x_3 - C_5(sign\ x_1)(x_1)^2 - C_3(sign\ \dot{x}_2)(\dot{x}_2)^2 +$
   $C_3(sign\ \dot{x}_2)(\dot{x}_2)^2 + C_3(sign\ \dot{x}_3)(\dot{x}_3)^2 + (C_4 + C_5)(sign\ x_3)(x_3)^2 = 0$

Initial Conditions

$x_1(0)=0.02\ m$ $x_2(0)=0.015\ m$ $x_3(0)=0.015\ m$

$\dot{x}_1(0)=0$ $\dot{x}_2(0)=0$ $\dot{x}_3(0)=0$

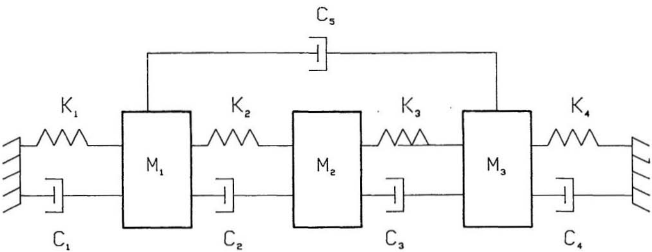


Fig. 2.2 Free Vibration of A Multi-Degree-System

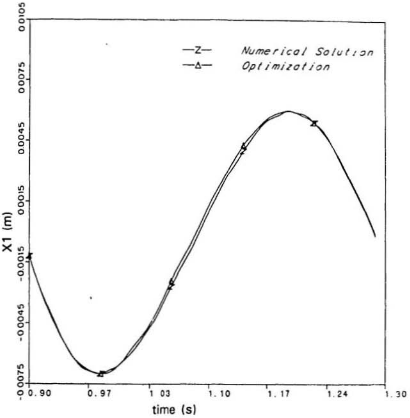


Fig. 2.3 Free Vibration Response of X_1 - Case No. 5

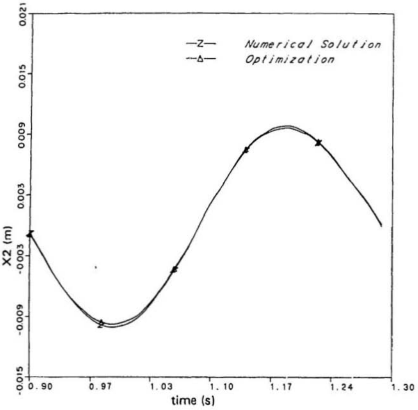


Fig. 2.4 Free Vibration Response of X_2 - Case No. 5

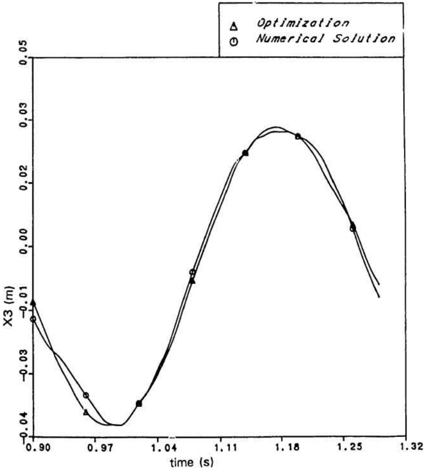


Fig. 2.5 Free Vibration Response of X_3 - Case No. 5

CHAPTER 3

BALANCING OF ROTORS SUPPORTED ON NON-LINEAR BEARINGS

3.1 Introduction

The optimization method of linearization developed in Chapter 2 shall now be used in the balancing of a rotor supported on non-linear bearings. The procedure is outlined on Fig. 3.1. The chapter is divided into the following sections:

1. Formulation of the equations of motion for the unbalanced response of a rotor supported on non-linear bearings.
2. Development of a numerical procedure to solve the non-linear equations.
3. Solution of a linear system of equations using the impedance method.
4. Equivalent linearization of the non-linear system using the optimization method.
5. Balancing of a rotor with non-linear bearings.

3.2 Derivation of Equations of Motion For A Rotor Supported on Non-linear Bearings

A multi-mass rotor mounted on non-linear ball bearings, one on each end of the shaft, is shown in Fig. 3.2. The equations of motion for this multi-degree of freedom system are formulated using flexibility influence coefficients and Lagrange's Equations (Tse et al, 1978-a). Using the Lagrange's approach the equations of motion are derived from

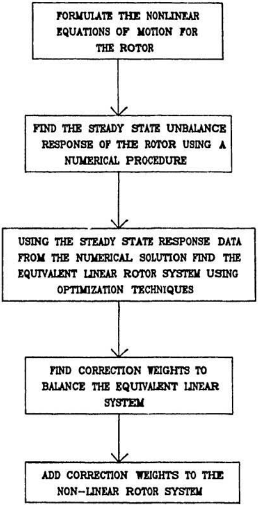


Fig. 3.1 Flow Chart of Balancing Procedure for Rotors Supported on Non-Linear Bearings

two scalar functions; the kinetic energy T, and the potential energy V. The equations apply to non-linear as well as linear systems.

The rotor consists of n number of disks of equal mass with a nominal mass assumed at the bearing locations. This mass was added at the bearing locations to eliminate the zero in the diagonal terms of the mass matrix $[M]$, in the derived equations of motion. It is assumed that the bearing mass and its associated kinetic energy is almost negligible. The mass of the shaft was neglected in this analysis, however, if it was to be included then it could have been replaced by masses at the bearings and disk locations. The rotor disks are assumed to be eccentric, with the location of the disk's centre of mass specified by the distance r_i and phase angle ϕ_i as shown in Fig. 3.2.

Referring to Fig. 3.2, the total rotor disk deflections are given by z_i and y_i, the displacements of the rotor disks due to the bearing displacements only, are given by z_i^* and y_i^*, while the elastic deformations along the z and y axes represented by vectors $\{u\}_z$ and $\{u\}_y$ are given by

$$\{u\}_z = \begin{Bmatrix} z_1 - z_1^* \\ z_2 - z_2^* \\ \cdot \\ z_n - z_n^* \end{Bmatrix} \qquad \{u\}_y = \begin{Bmatrix} y_1 - y_1^* \\ y_2 - y_2^* \\ \cdot \\ y_n - y_n^* \end{Bmatrix} \tag{3.1}$$

The potential energy of the rotor comprises of the strain energy from both bearing deflections and rotor elastic deformations. The non-linear bearings are modelled as ball

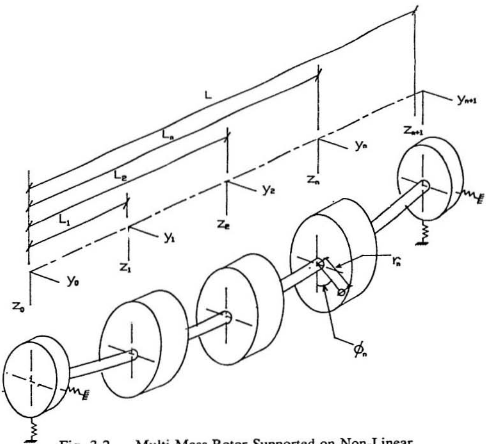


Fig. 3.2 Multi-Mass Rotor Supported on Non-Linear Bearings

bearings, such that they have a cubic non-linear term (Genta and Repaci,1988), where the force and displacement expressions are given by

$$
\begin{aligned}
F_{z_0} &= K_{zz}^{1B} z_0 + \varepsilon^{1B} z_0^3 \\
F_{z_{n+1}} &= K_{zz}^{2B} z_{n+1} + \varepsilon^{2B} z_{n+1}^3 \\
F_{y_0} &= K_{yy}^{1B} y_0 + \varepsilon^{1B} y_0^3 \\
F_{y_{n+1}} &= K_{yy}^{2B} y_{n+1} + \varepsilon^{2B} y_{n+1}^3
\end{aligned}
\tag{3.2}
$$

The potential energy for the bearings is given by

$$
\begin{aligned}
V_B = & \int_0^{z_0} (K_{zz}^{1B} z + \varepsilon^{1B} z^3)\, dz + \\
& \int_0^{y_0} (K_{yy}^{1B} y + \varepsilon^{1B} y^3)\, dy + \\
& \int_0^{z_{n+1}} (K_{zz}^{2B} z + \varepsilon^{2B} z^3)\, dz + \\
& \int_0^{y_{n+1}} (K_{yy}^{2B} y + \varepsilon^{2B} y^3)\, dy
\end{aligned}
\tag{3.3}
$$

The potential energy of the rotor due to elastic deformations is given by

$$
V_S = \frac{1}{2} \{u\}_z^T [K_s] \{u\}_z + \frac{1}{2} \{u\}_y^T [K_s] \{u\}_y \tag{3.4}
$$

where $[K_s]$ is the matrix of stiffness influence coefficients for the shaft. The matrix $[K_s]$ can be found by taking the inverse of the flexibility influence coefficients. (See Appendix [B] for the derivation of the shaft stiffness matrix, $[K_s]$). The total potential energy V_T is the sum of V_B and V_S such that

$$
V_T = V_B + V_S \tag{3.5}
$$

Eq. (3.5) does not include a term due to the weight of the rotor disks. This is because the static deflections are not a function of time. In order to find the total deflection, one must separately calculate the static deflection due to gravity and use the superposition principle for obtaining the total deflection.

The expression for the system kinetic energy is given by

$$T = \sum_{i=1}^{n} \frac{1}{2} m_i [\frac{d}{dt}(z_i + r_i \cos(\omega t + \phi_i))]^2 + \sum_{i=1}^{n} \frac{1}{2} m_i [\frac{d}{dt}(y_i + r_i \sin(\omega t + \phi_i))]^2 \tag{3.6}$$

Having found the kinetic and potential energy expressions, we can introduce the Lagrangian, defined by:

$$L^{*} = T - V_T \tag{3.7}$$

The Lagrange's equations of motion (Tse et al, 1978-a), can be written as

$$\frac{d}{dt}(\frac{\partial L^{*}}{\partial \dot{q}_i}) - \frac{\partial L^{*}}{\partial q_i} + \frac{\partial \psi}{\partial \dot{q}_i} = F_i \qquad i=1...n \tag{3.8}$$

In Eq. (3.8) F_i represents the nonconservative forces, q_i the generalized coordinate, and ψ represents the Rayleigh's dissipation function. In this rotor example, there is no viscous damping and therefore $d\psi/d\dot{q}_i = 0$. The rigid body mode displacements z_i^{*} and y_i^{*} can be eliminated from Eq. (3.4) using the following relationships

$$z_i^{*} = z_0 + l_i \tan\theta_z = z_0(1 - \frac{l_i}{L}) + (\frac{l_i}{L})z_{n+1} \tag{3.9}$$

$$y_i^* = y_0 + l_i \tan\theta_y = y_0\left(1 - \frac{l_i}{L}\right) + \left(\frac{l_i}{L}\right) y_{n+1}$$

where $\theta_z = \dfrac{z_{n+1} - z_0}{L}$ *and* $\theta_y = \dfrac{y_{n+1} - y_0}{L}$ (3.9a)

Substituting Eqs. (3.5), (3.6), and (3.9) into Eq. (3.8), one obtains the dynamic equations of motion of the system as:

$$[M] \begin{Bmatrix} \ddot{z}_0 \\ \ddot{y}_0 \\ \vdots \\ \ddot{z}_{n+1} \\ \ddot{y}_{n+1} \end{Bmatrix} + [K] \begin{Bmatrix} z_0 \\ y_0 \\ \vdots \\ z_{n+1} \\ y_{n+1} \end{Bmatrix} = \{F\} \tag{3.10}$$

where $\{F\}$ is the unbalance force vector given as

$$\{F\} = \begin{Bmatrix} 0 \\ 0 \\ m_2 r_2 \omega^2 \cos(\omega t + \phi_2) \\ m_2 r_2 \omega^2 \sin(\omega t + \phi_2) \\ \cdot \\ \cdot \\ m_5 r_5 \omega^2 \cos(\omega t + \phi_5) \\ m_5 r_5 \omega^2 \sin(\omega t + \phi_5) \\ 0 \\ 0 \end{Bmatrix} \tag{3.11}$$

See Appendix [C] for the details of the derivation and the resulting equations of motion and various matrices.

Introducing a structural damping matrix $[C]$ which is also shown in detail in Appendix B, and using the generalized coordinate q_i, the final set of equations becomes

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{F\} \tag{3.12}$$

where

$$\{q\} = \begin{Bmatrix} z_0 \\ y_0 \\ z_1 \\ y_1 \\ \cdot \\ \cdot \\ z_{n+1} \\ y_{n+1} \end{Bmatrix} \tag{3.13}$$

The overall system stiffness matrix $[K]$ contains the bearing non-linear stiffness terms expressed in Eq. (3.2) as follows:

$$\begin{matrix} K_{zz}^{1B} + \epsilon^{1B} z_0^2 \\ K_{yy}^{1B} + \epsilon^{1B} y_0^2 \\ K_{zz}^{2B} + \epsilon^{2B} z_{n+1}^2 \\ K_{yy}^{2B} + \epsilon^{2B} y_{n+1}^2 \end{matrix} \tag{3.14}$$

3.3 Numerical Solution of the Non-Linear System

The numerical time response solution for the non-linear system of equations is calculated using the Newmark-β Method (Rao, 1986) in conjunction with an iteration procedure. From the Newmark-β Method the resulting equations for the velocity and displacement vectors can be expressed as:

$$\{\dot{q}\}_{k+1} = \{\dot{q}\}_k + [(1-\beta)\{\ddot{q}\}_k + \beta\{\ddot{q}\}_{k+1}]\Delta t, \quad and \tag{3.15}$$

$$\{q\}_{k+1} = \{q\}_k + \Delta t\{\dot{q}\}_k + [(0.5-\alpha)\{\ddot{q}\}_k + \alpha\{\ddot{q}\}_{k+1}]\Delta t^2 \tag{3.16}$$

where k is the iteration step in time, and the parameters α and β indicate how much acceleration, at the end of interval Δt, enters into the velocity and displacement equations. To find the value of $\{\ddot{q}\}_{k+1}$, the equation of motion (Eq. 3.12) for $t = t_{k+1}$ is given by

$$[M]\{\ddot{q}\}_{k+1} + [C]\{\dot{q}\}_{k+1} + [K]\{q\}_{k+1} = \{F\}_{k+1} \tag{3.17}$$

We want to substitute $\{\ddot{q}\}_{k+1}$ and $\{\dot{q}\}_{k+1}$ in terms of $\{q\}_{k+1}$ in Eq. (3.17). All the vectors at the time instant k in Eq. (3.15) and Eq. (3.16) are known from the initial conditions. One can express $\{\ddot{q}\}_{k+1}$ in terms of $\{q\}_{k+1}$ in Eq. (3.16) and substituting $\{\ddot{q}\}_{k+1}$ in terms of $\{q\}_{k+1}$ in Eq. (3.15), one can find out $\{\dot{q}\}_{k+1}$ in terms of $\{q\}_{k+1}$. In this way we can rewrite Eq. (3.17) with $\{q\}_{k+1}$ as the unknown, which is written as:

$$\begin{aligned}\{q\}_{k+1} &= [(\frac{1}{\alpha(\Delta t)^2})[M] + (\frac{\beta}{\alpha \Delta t})[C] + [K]_{k+1}]^{-1} \\ &\times \Big(\{F\}_{k+1} + [M](\frac{1}{\alpha(\Delta t)^2}\{q\}_k + \frac{1}{\alpha \Delta t}\{\dot{q}\}_k + (\frac{1}{2\alpha}-1)\{\ddot{q}\}_k) \\ &+ \left([C]\left\{\frac{\beta}{\alpha \Delta t}\{q\}_k + (\frac{\beta}{\alpha}-1)\{\dot{q}\}_k + (\frac{\beta}{\alpha}-2)\frac{\Delta t}{2}\{\ddot{q}\}_k\right\}\right)\end{aligned} \tag{3.18}$$

In the non-linear rotor's case, the stiffness matrix $[K]_{k+1}$ in Eq. (3.18) has non-linear terms which depend on the values of the elements in vector $\{q\}_{k+1}$. Eq. (3.18) is a set of non-linear algebraic equations now. Therefore an iteration procedure is utilized in conjunction with the Newmark-β Method to find $\{q\}_{k+1}$. The following steps describe the numerical procedure:

1. Specify initial conditions at $t = 0$, $\{q\}_0$ and $\{\dot{q}\}_0$.

2. Using the known initial conditions from Step (1) and Eq. (3.12), $\{\ddot{q}\}_0$ is calculated as

$$\{\ddot{q}\}_0 = [M]^{-1}[\{F_{t=0}\} - [C]\{\dot{q}\}_0 - [K]\{q\}_0 \tag{3.19}$$

where some of the terms of $[K]$ are a function of $\{q\}_0$.

3. Select the time step Δt, and parameters α and β.

4. Starting with $k = 0$, assume a displacement vector $(\{q\}_{k+1})_A$.

5. Calculate the non-linear terms of the stiffness matrix $[K]$, using values from the assumed displacement vector $(\{q\}_{k+1})_A$ in Eq. (3.13). Assemble the stiffness matrix $[K]$.

6. Calculate displacement vector $\{q\}_{k+1}$ using Eq. (3.18) and the stiffness matrix $[K]$ from Step 5.

7. Compare $\{q\}_{k+1}$ with assumed displacement vector. If the difference is not within specified tolerances, use an average value of the assumed displacement vector from Step (4) and the calculated displacement vector from Step (6) for the new assumed vector , and return to Step (4). If the difference is within tolerance then proceed to Step (8).

8. Calculate the acceleration and velocity at t_{k+1} from the following expressions which are slightly modified forms of Eqs. (3.17) and (3.16).

$$\begin{aligned} \{\ddot{q}\}_{k+1} &= \frac{1}{\alpha \Delta t^2}(\{q\}_{k+1} - \{q\}_k) - \frac{1}{\alpha \Delta t}\{\dot{q}\}_k - (\frac{1}{2\alpha} - 1)\{\ddot{q}\}_k \\ \{\dot{q}\}_{k+1} &= \{\dot{q}\}_k + (1-\beta)\Delta t\{\ddot{q}\}_k + \beta \Delta t\{\ddot{q}\}_{k+1} \end{aligned} \tag{3.20}$$

(3.21)

9. Update the assumed vector for t_{k+2} and go to Step (4). Continue for $k = N$ iterations to obtain the steady state solution.

3.4 Solution of Linear Systems Using the Impedance Method

In Section 3.3 we obtained the numerical solution of the nonlinear system. As was done in Chapter 2, we will find an equivalent linear system using the optimization principles and then balance the equivalent linear system with a set of weights. Before we find the equivalent system it is worth briefly going over the impedance method (Tse et al, 1978-b) for finding the response of the linear system. Moreover, even to find the equivalent linear system one needs to know the linear response equations. The equations of motion for a linearized system are given by

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K_{Eq}]\{q\} = \{F\} \tag{3.22}$$

where the matrix $[K_{Eq}]$ is the equivalent linear stiffness matrix, and $[M]$, $[C]$, and $\{F\}$ matrices are unchanged from the original non-linear equations (Eq. (3.12)).

In the case of the rotor, the excitation force vector from the inherent unbalance is harmonic and all the elements in $\{F\}$ are of the same frequency. Therefore $\{F\}$ is expressed as

$$\{F\} = \{\bar{F}e^{j\omega t}\} = \{\bar{F}\}e^{j\omega t} \tag{3.23}$$

where $\{\bar{F}\}$ is the complex phasor of $\{F\}$. If we let the harmonic response be $\{Q\}$ where

$$\{Q\} = \{\bar{Q}e^{j\omega t}\} = \{\bar{Q}\}e^{j\omega t} \tag{3.24}$$

and $\{\bar{Q}\}$ is the complex phasor of $\{Q\}$; applying the impedance method and factoring out $e^{j\omega t}$, Eq. (3.22) becomes

$$-\omega^2[M]\{\bar{Q}\} + j\omega[C]\{\bar{Q}\} + [K_{Eq}]\{\bar{Q}\} = \{\bar{F}\} \tag{3.25}$$

or

$$[\,[K_{Eq}] - \omega^2[M] + j\omega[C]\,]\{\bar{Q}\} = \{\bar{F}\} \tag{3.26}$$

$$[Z]\{\bar{Q}\} = \{\bar{F}\}$$

$$\{\bar{Q}\} = [Z]^{-1}\{\bar{F}\} \tag{3.27}$$

where $[Z]$ is the impedance matrix, and $\{\bar{Q}\}$ is the solution vector, which gives the amplitude and phase angle of the response. From the harmonic unbalance force vector $\{F\}$, the elements of the complex force vector $\{\bar{F}\}$ in Eq. (3.27) are given by (See Appendix E):

$$\begin{aligned}
&\bar{F}(1)=0\\
&\bar{F}(2)=0\\
&\bar{F}(3)=(m_2r_2\omega^2\cos\phi_2)+(m_2r_2\omega^2\sin\phi_2)j\\
&\bar{F}(4)=(m_2r_2\omega^2\cos(\phi_2+\frac{3\pi}{2}))+(m_2r_2\omega^2\sin(\phi_2+\frac{3\pi}{2}))j\\
&\bar{F}(5)=(m_3r_3\omega^2\cos\phi_3)+(m_3r_3\omega^2\sin\phi_3)j\\
&\bar{F}(6)=(m_3r_3\omega^2\cos(\phi_3+\frac{3\pi}{2}))+(m_3r_3\omega^2\sin(\phi_3+\frac{3\pi}{2}))j\\
&\bar{F}(7)=(m_4r_4\omega^2\cos\phi_4)+(m_4r_4\omega^2\sin\phi_4)j\\
&\bar{F}(8)=(m_4r_4\omega^2\cos(\phi_4+\frac{3\pi}{2}))+(m_4r_4\omega^2\sin(\phi_4+\frac{3\pi}{2}))j\\
&\bar{F}(9)=(m_5r_5\omega^2\cos\phi_5)+(m_5r_5\omega^2\sin\phi_5)j\\
&\bar{F}(10)=(m_5r_5\omega^2\cos(\phi_5+\frac{3\pi}{2}))+(m_5r_5\omega^2\sin(\phi_5+\frac{3\pi}{2}))j\\
&\bar{F}(11)=0\\
&\bar{F}(12)=0
\end{aligned}$$

3.5 Equivalent Linearization Using the Optimization Principle

Once the non-linear steady state solution is found, we can find the equivalent linear system using the optimization method outlined in Chapter 2. An equivalent linear stiffness matrix $[K_{Eq}]$ will be found to replace the non-linear stiffness matrix $[K]$ such that the error in the rotor amplitudes between the numerical solution and the equivalent linear system solution is minimized. The following steps are used to obtain the values of the $[K_{Eq}]$ matrix:

1. Select a period of the steady state solution from the numerical solution, and obtain vector $\{q(t)\}_{NUM}$.

2. Assume a vector of whose components are the bearing equivalent linear stiffness values within specified upper and lower ranges.

$$\{V\} = \begin{Bmatrix} K^{1B}_{zz_{Eq}} \\ K^{1B}_{yy_{Eq}} \\ K^{2B}_{zz_{Eq}} \\ K^{2B}_{yy_{Eq}} \end{Bmatrix} \tag{3.28}$$

3. Replace the non-linear stiffness terms shown in Eq. (3.14) with the assumed values from vector $\{V\}$ and calculate the equivalent linear stiffness matrix $[K_{Eq}]$. Obtain the steady state response for each degree of freedom for the equivalent linear system using the impedance method Eq. (3.27), outlined in the previous section of this chapter. Using $\{\bar{Q}\}$, the response vector for the equivalent linear system $\{q(t)\}_{Eq}$ can be found using Eq.(3.24).

4. Calculate the value of the objective function by defining the error as

$$e_j = \{q(t)\}_{NUM} - \{q(t)\}_{Eq} \tag{3.29}$$

The objective function becomes

$$U = \sum_{j=1}^{N} e_j^2 \quad where\ j = 1,N \tag{3.30}$$

5. Follow the steps in the minimization algorithm outlined in Chapter 2 to find the equivalent linear stiffness values in vector $\{V\}$.

3.6 Balancing

3.6.1 Introduction - Rotor Balancing

The balancing of rotors is accomplished by adding compensating unbalances at discrete rotor locations, or by machining. Some rotors can be considered "rigid". That is, the shape of the rotor and shaft are invariable, even at high rotational speeds. For these types of rotors, static balancing and dynamic balancing techniques can be used to find the correction weights.

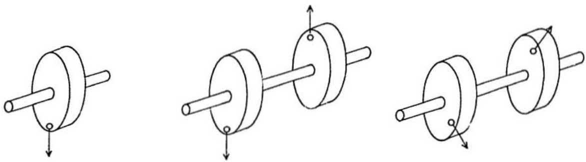
If we have a uniform thin disk of mass m, with eccentricity r, the correction mass m_{cor} required at the radial distance r_m is

$$m_{cor} = m\frac{r}{r_m} \tag{3.31}$$

The location where the correction mass can be added or removed can be found by simple statics. The rotor can be placed on two horizontal knife edges and allowed to roll freely until the heavy spot comes to the lowest position. This is referred to as static balancing or single plane balancing (See Fig. 3.3(a)). Static balancing requires a correction mass only on one plane.

In many cases, the unbalance is distributed along the length of the rotor and static balancing cannot be used to determine the correction masses. As shown in Fig. 3.3(b), it is possible to have two equal uncompensated masses symmetrically placed about the centre mass, but positioned at 180 degrees to one another. The rotor is in static balance, yet centrifugal forces will produce a moment about the centre of mass when the rotor turns. This type of unbalance is called couple unbalance, and results in a tilting or pendulum action of the principle inertia axis about the shaft axis at the centre of mass. To counteract the couple it is necessary to make corrections on two planes.

The general case where both static and couple unbalance exist, is called dynamic unbalance (Fig. 3.3(c)). The principle inertia axis is now inclined to the geometric shaft axis, but there is also an eccentricity at the centre of the mass. A rigid rotor with dynamic unbalance can be balanced by adding correction weights on any two separate planes normal to the shaft axis.



a) Static Unbalance b) Couple Unbalance c) Dynamic Unbalance

Fig. 3.3 Types of Rotor Unbalance

When the rotor is rigid, ie. the shape of the centroidal axis does not change as a function of speed, the dynamic balancing of the rotor is quite simple. However, in most high speed applications, rotors operate above the first critical speed and can be no longer considered as rigid rotors. The rotor deflects and the shaft's bent centerline whirls around and additional centrifugal forces are set up. For these rotors, flexible balancing techniques are required. The two principle methods for flexible rotor balancing are the modal balancing method and the influence coefficient method. In this work the influence coefficient method will be used. Either method could have been used in this study, however the least squares influence coefficient method was chosen because it is the most commonly used method with the most practical applications. Modal balancing is not a practical method due to the large amplitudes at the critical speeds. A comprehensive discussion of modal balancing theory is provided by Darlow (1989).

3.6.2 Balancing of Linear Systems Using The Least Squares Influence Coefficient Method

As was discussed in Chapter 1, the influence coefficient method utilizes the rotor-bearing sensitivity for the calculation of balance correction weights (Rao, 1983). For linear systems this sensitivity can be expressed in matrix form, where the complex elements of which are termed influence coefficients. The influence coefficients are obtained by determining the rotor response to known trial masses at particular balancing planes for a selected balancing speed. Assume we have collected this data for the z direction shaft displacements as given in Fig. 3.2. These rotor responses are compared to the uncorrected or original unbalanced rotor response data to determine the influence coefficients, mathematically expressed as

$$a_{ij} = \frac{(z_{ij} - z_{i0})}{T_j} \tag{3.32}$$

where a_{ij} is the complex influence coefficient which is also an element of the matrix [A] discussed in the subsequent pages, z_{io} is the i^{th} vibration reading with no trial masses installed in the j^{th} balancing plane, z_{ij} is the i^{th} vibration reading with the trial mass installed in the j^{th} balancing plane, and T_j is the trial mass installed, expressed as a complex value representing its magnitude and angular location on the balancing j^{th} plane. In addition to the initial uncorrected rotor data, for b balancing planes, b runs are necessary to collect all the data to calculate the influence coefficient matrix. For this discussion we will assume that there are p vibration measurements taken along the rotor for each run.

Once the influence coefficient matrix is obtained, then the correction masses $\{m_c\}$ are calculated. The residual vibration $\{z\}$, i.e. the vibration of the rotor with the trial masses installed, is expressed as

$$\{z\} = \{z_0\} + \{z_m\} \tag{3.33}$$

where $\{z_o\}$ is the response due to the original rotor unbalance expressed as complex numbers, and $\{z_m\}$ is the vector of amplitudes with the correction masses installed. Using the influence coefficient matrix found from the trial masses, $\{z_m\}$ is expressed as

$$\{z_m\} = [A]\{m_c\} \tag{3.34}$$

Substituting Eq. (3.34) into Eq. (3.33) the residual vibration is

$$\{z\}_{px1} = \{z_0\}_{px1} + [A]_{pxb}\{m_c\}_{bx1} \tag{3.35}$$

When the number of balancing planes b, and the number of measurement planes p are equal, Eq. (3.33) can be solved directly to reduce all values of $\{z\}$ to zero to give

$$\{m_c\} = -[A]^{-1}\{z_0\} \tag{3.36}$$

where $[A^{-1}]$ is the inverse of the square influence coefficient matrix, and $\{m_c\}$ is the complex vector representing the correction masses and the angular locations.

If the number of measurements p is greater then the number of balancing planes, then we cannot in general reduce all residual vibrations to zero, but we can minimize the sum of their squares. Expressing the sum of the squares of the residual vibration as S, it is as follows

$$S = \sum_{i=1}^{p} |z_i|^2 = \sum_{i=1}^{p} \bar{z}_i z_i \tag{3.37}$$

where $\bar{z}_i$ are the complex conjugates of the elements z_i. S is minimized by

$$\frac{\partial S}{\partial \bar{m}_{c_j}} = 0 = \sum_{i=1}^{p} z_i \frac{\partial \bar{z}_i}{\partial \bar{m}_{c_j}} = \sum_{i=1}^{p} z_i \bar{a}_{i,j} \qquad (j=1...b) \tag{3.38}$$

where $\bar{m}_{cj}$ are the complex conjugates of the elements of the vector $\{m_c\}$, and $\bar{a}_{i,j}$ are the complex conjugates of the elements of the matrix $[A]$. The expression for the residual vibration (Eq 3.33) can be rewritten as

$$z_i = z_{0_i} + \sum_{j=1}^{b} a_{ij} m_{c_j} \qquad (i=1...p) \tag{3.39}$$

where z_{oi} are the elements of $\{z_o\}$. Substituting Eq. (3.37) into Eq. (3.36), we obtain

$$\sum_{i=1}^{p} \bar{a}_{ij}[z_{0_i} + \sum_{f=1}^{b} a_{if} m_{c_f}] = 0 \qquad (j=1...b) \tag{3.40}$$

rearranging this expression in matrix form yields

$$-[\bar{A}]^T[A]\{m_c\} + -[\bar{A}]^T\{z_0\} = 0 \tag{3.41}$$

where $[A]^T$ is a conjugate transpose of $[A]$ and $[\bar{A}]$ is the complex conjugate of $[A]$. This equation can be solved to give

$$\{m_c\} = -([\bar{A}]^T[A])^{-1}[\bar{A}]^T\{z_0\} \tag{3.42}$$

This is the set of correction masses which minimizes the residual vibration of the linear rotor system.

3.6.3 Balancing of a Rotor Supported on Non-Linear Bearings

The balancing procedure discussed in the previous section applies to linear systems. In precise balancing work with rotors with non-linearities, i.e. non-linear supports, the non-linearities have to be considered. This can be accomplished by using equivalent linearization. Once the non-linear equations of motion are formulated, an equivalent linear system is obtained for the balancing speed of interest for use in the balancing program. The equivalent linear system is found by using the optimization techniques discussed in Section 3.5. This linear system is used to calculate the rotor response to the unbalance forces near it's critical speed. The least square influence coefficient method of balancing is then used to find the correction weights to reduce the equivalent linear rotor's response to acceptable levels. These correction weights are used to balance the original non-linear rotor.

This balancing procedure can be used in conjunction with computational rotor models in which one would like to consider the effect of the rotor non-linearities. This could be done at the design stage where the rotor response and unbalance sensitivity studies are carried out to verify the performance and integrity of new rotor designs. Also, computational models can be used to aid in the balancing of rotors in the field. The availability of an accurate model will enable the unbalance to be estimated from the displacement data collected, thus balancing can be carried out without the need for numerous trial runs to estimate the influence coefficient matrix. Again, this balancing procedure will enable incorporation of rotor non-linearities.

3.7 Numerical Examples

3.7.1 Rotor Model

To illustrate the balancing procedure a rotor bearing system with four disks was chosen. A negligible bearing mass was included in the model. Fig. 3.4 shows the rotor configuration while Table 3.1 gives the parameters of the three different rotors

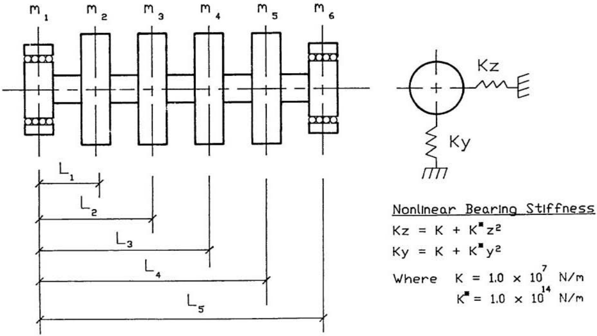


Fig. 3.4 Rotor System

Table 3.1 Rotor Parameters

Rotor Parameters	Rotor No. 1	Rotor No. 2	Rotor No. 3
Shaft Diameter	0.050 m	0.076 m	0.089 m
M1	0.10 kg	0.05 kg	0.05 kg
M2	30.0 kg	20.0 kg	20.0 kg
M3	30.0 kg	20.0 kg	20.0 kg
M4	30.0 kg	20.0 kg	20.0 kg
M5	30.0 kg	20.0 kg	20.0 kg
M6	0.10 kg	0.05 kg	0.05 kg
L1	0.127 m	0.20 m	0.30 m
L2	0.254 m	0.45 m	0.55 m
L3	0.508 m	0.75 m	0.75 m
L4	0.635 m	1.00 m	1.10 m
L5	0.760 m	1.20 m	1.40 m
	L/D = 15.2	L/D = 15.79	L/D = 15.73

considered. Rotor No.1 consists of a shaft, 0.050 m in diameter, 0.760 m in length, with (4) four 30 kg masses and (2) two 0.10 kg masses at the bearings. The shaft diameter and length was increased on Rotor No.2, and the disk masses decreased to 20 kg. Rotor No.3 has the same disk masses as Rotor No. 2, but the rotor length and shaft diameter were increased. For all three rotors the ratio L/D was approximately maintained (15.2,15.8,15.7).

The material density of 7,832 kg/m^3 and an elastic modulus of 200 x 10^9 N/m^2 was used to model the rotor shaft. The bearing linear stiffness coefficient was chosen as 1.0 x 10^{07} N/m while the non-linear bearing stiffness coefficient was selected as 1.0 x 10^{14} N/m.

The unbalance condition assumed for each of the four disks was an eccentricity of 0.00002 m, with m_2 and m_3 in phase with a phase angle of 0.785 radians, and m_4 and m_5 in phase with a phase angle of 2.355 radians.

3.7.2 Numerical Solution

The solution technique used for the numerical solution of the equations of motion was the Newmark-β Method in conjunction with an iteration procedure as described earlier in this chapter. The time step chosen was 0.0001 seconds. The constants α and β mentioned in Eq. (3.18) were chosen as 0.25 and 0.50 respectively to give a numerically stable response (Rao, 1986-b). The total number of iterations were chosen such that the plotted results represented acceptable steady-state conditions.

This method consumed large amounts of computer time, since the steady state amplitudes for a large number of speeds were required to generate a speed versus amplitude curve as shown in Fig. 3.5. This figure shows the speed versus amplitude curve for z_2 of Rotor No.1. Sufficient computer time must be allowed at each speed for the initial transients to die out. In order to minimize this time the damping coefficient in the structural damping matrix was increased to 0.05. Despite this high assumed damping a typical calculation for the steady state rotor response at one speed took just under 2 hours CPU time, with a total elapsed time of approximately 4 hours. This time increased when the speed selected approached the critical speed. The speed versus steady state amplitude curves for the original unbalanced condition for the three rotors are shown in Appendix D. The steady state amplitudes with the balance weights added, were calculated for the selected balancing speeds only. These results are discussed in Section 3.7.4.

3.7.3 Equivalent Linearization

In the optimization method of linearization, one cycle of the steady state response of each degree of freedom of the rotor disks were selected, at a speed near the critical speed. This data became the "exact solution". There were 501 data points (t versus amplitude) for each of the degrees of freedom.

The optimization problem parameters were chosen. The maximum and minimum values of the equivalent linear bearing stiffness were chosen as 5.0×10^7 N/m and 1.0

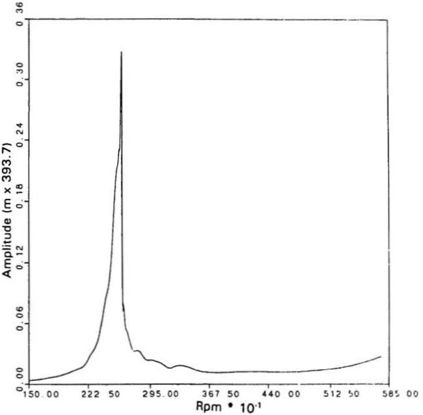


Fig. 3.5 Rotor Unbalance Response - Typical

$\times 10^7$ N/m respectively. Starting values for the four stiffness values were each selected within these margins. If the solution failed to converge after 50 runs, then the procedure was stopped. The convergence was sensitive to the selected starting values. Table 3.2 shows the equivalent linear stiffness values calculated for each of the rotors.

3.7.4 Balancing

Once the rotor equivalent linear bearing stiffness values were determined, the least squares influence coefficient balancing procedure was used to determine the correction weights. These weights were then added to the non-linear rotor system to reduce the rotor response.

The number of rotor balancing planes b was chosen as 3, while the number of measuring planes was selected as 6 (See Fig. 3.3). The vertical rotor displacements only were used in the balancing data. The rotor displacements were calculated using the impedance method discussed in Section 3.4. They were calculated at the first natural frequency for the equivalent linear system, which differed from the natural frequencies observed from the rotor amplitude curves for the non-linear rotors. From the impedance solution, the complex rotor amplitudes at the selected measurement planes were obtained for the original unbalanced condition. Then with a trial weight added to a balancing plane, three additional sets of rotor amplitudes were found. From this data, the matrix of influence coefficients were calculated as per Eq. (3.32). As an example, Table 3.3 shows the balancing data obtained for Rotor No. 2. Using Eq. (3.42) a set of correction weights

Table 3.2 Equivalent Linear Bearing Stiffness

	Rotor No. 1	Rotor No. 2	Rotor No. 3
$(K_{z1})_{Eq}$	1.0000 X 10^7	1.2121	1.0400
$(K_{y1})_{Eq}$	1.0272 X 10^7	1.2109	1.0478
$(K_{z2})_{Eq}$	1.5106 X 10^7	1.0650	1.3171
$(K_{y2})_{Eq}$	1.4181 X 10^7	1.0650	1.2975
SPEED	2550	3270	3250

TABLE 3.3 BALANCING DATA - ROTOR NO. 2

BALANCING DATA - ROTOR NO. 2											
ORIGINAL UNBALANCE			TRIAL WT. NO. 1			TRIAL WT. NO. 2			TRIAL WT. NO. 3		
Plane	Kg*m	Radians	Plane	Kg*m	Radians	Plane	Kg*m	Radians	Plane	Kg*m	Radians
2 and 3	.0004	0.785	2	.0009	0.785	3	.0009	4.713	5	.0009	2.356
4 and 5	.0004	2.355									
COMPLEX AMPLITUDES (m)			COMPLEX AMPLITUDES (m)			COMPLEX AMPLITUDES (m)			COMPLEX AMPLITUDES (m)		
z_{10}	$6.29X10^{-4}$	$3.09x10^{-7}$	z_{11}	$9.19X10^{-4}$	$-2.90X10^{-4}$	z_{12}	$4.53X10^{-5}$	$6.99X10^{-6}$	z_{13}	$9.41X10^{-4}$	$2.99X10^{-4}$
z_{20}	$1.27X10^{-3}$	$-1.32X10^{-6}$	z_{21}	$1.85X10^{-3}$	$-5.92X10^{-4}$	z_{22}	$8.81X10^{-5}$	$1.43X10^{-5}$	z_{23}	$1.89X10^{-3}$	$6.02X10^{-4}$
z_{30}	$1.81X10^{-3}$	$-3.25X10^{-6}$	z_{31}	$2.63X10^{-3}$	$-8.50X10^{-4}$	z_{32}	$1.23X10^{-4}$	$2.08X10^{-5}$	z_{33}	$2.70X10^{-3}$	$8.60X10^{-4}$
z_{40}	$1.83X10^{-3}$	$-4.45X10^{-6}$	z_{41}	$2.66X10^{-3}$	$-8.66X10^{-4}$	z_{42}	$1.22X10^{-4}$	$2.29X10^{-5}$	z_{43}	$2.74X10^{-3}$	$8.72X10^{-4}$
z_{50}	$7.33X10^{-3}$	$-3.91X10^{-6}$	z_{51}	$1.93X10^{-3}$	$-6.32X10^{-4}$	z_{52}	$8.70X10^{-5}$	$1.82X10^{-5}$	z_{53}	$1.99X10^{-3}$	$6.35X10^{-4}$
z_{60}	$7.21X10^{-4}$	$-2.72X10^{-6}$	z_{61}	$1.05X10^{-3}$	$-3.45X10^{-4}$	z_{62}	$4.55X10^{-5}$	$1.13X10^{-5}$	z_{63}	$1.08X10^{-3}$	$3.47X10^{-4}$

Note: Speed = 350.3 Rad/s

which minimized the unbalance responses were obtained. These correction weights were then added to the non-linear rotors and the rotor response was calculated. Tables 3.4 to 3.6 show the reduction in the unbalance response near the first critical speed and one other higher operating speed for each of the non-linear rotors. These results clearly show that one can balance the non-linear system by first finding an equivalent linear system and then the balancing weights for this linear system.

3.8 Conclusions

In this chapter, the equations of motion were derived using the Lagrange Equations and the influence coefficient method. The set of non-linear equations were solved using the Newmark- β time marching scheme in conjunction with an iteration procedure. The equivalent linear system was obtained using the optimization principle discussed in Chapter 2. The balancing weights for this linear system were obtained using the least squares influence coefficient matrix. The peak response values of the linear system were at different frequencies from the non-linear system. The balancing had to be done at the frequency which corresponded to the peak of the linear system. These weights were then used on the non-linear system and the results obtained were quite good. As a point of clarification, it should be mentioned here, that the majority of the computer time is consumed in finding the steady state response of the non-linear rotor system, and not in the equivalent linearization process.

Table 3.4 Rotor Response - Rotor No.1

RPM	Mass Station	Unbalanced Response (m x 39.37)	After First Balancing Run (m x 39.37)	% Reduction
2300	2	0.002576	0.000139	94.6
	3	0.003518	0.000153	95.6
	4	0.003515	0.000070	98.0
	5	0.002571	0.000033	98.7
2610	2	0.02237	0.000086	99.6
	3	0.03200	0.000167	99.5
	4	0.03163	0.000279	99.1
	5	0.02173	0.000247	98.9
5500	2	0.002716	0.001471	45.8
	3	0.002065	0.000955	53.7
	4	0.001686	0.000997	40.9
	5	0.002422	0.001533	36.7

Table 3.5 Rotor Response - Rotor No.2

RPM	Mass Station	Unbalanced Response (m x 39.37)	After First Balancing Run (m x 39.37)	% Reduction
3000	2	0.002535	0.000018	99.3
	3	0.003482	0.000037	98.9
	4	0.003485	0.000105	96.9
	5	0.002539	0.000120	95.3
3300	2	0.011510	0.000205	98.2
	3	0.016260	0.000203	98.7
	4	0.016232	0.000117	99.3
	5	0.011445	0.000034	99.7
5000	2	0.000964	0.000272	71.8
	3	0.001205	0.000149	87.6
	4	0.001164	0.000106	90.9
	5	0.000890	0.000243	72.7

Table 3.6 Rotor Response - Rotor No.3

RPM	Mass Station	Unbalanced Response (m x 39.37)	After First Bal.Run (m x 39.37)	% Red.	After Second Bal. Run (m x 39.37)	% Red.
3250	2	0.011150	0.001732	84.5	0.000444	96.0
	3	0.014063	0.002096	85.1	0.000519	96.3
	4	0.014400	0.002110	85.3	0.000508	96.5
	5	0.010828	0.001581	85.4	0.000361	96.7
3400	2	0.009638	0.000324	96.7	0.000117	96.6
	3	0.012455	0.000349	97.2	0.000110	99.1
	4	0.012793	0.000319	97.5	0.000085	99.3
	5	0.009472	0.000208	97.8	0.000032	99.7
5000	2	0.000924	0.000191	99.3	0.000116	87.4
	3	0.001272	0.000107	91.6	0.000056	95.6
	4	0.001285	0.000022	98.3	0.000018	98.6
	5	0.000981	0.000095	90.3	0.000100	90.3

CHAPTER 4

SUMMARY OF THE WORK AND RECOMMENDATIONS FOR FUTURE WORK

4.1 Summary

The method of balancing of multi-disk-rotor systems was developed in this work. The rotors were assumed to be supported on bearings with non-linear flexibility. At first, several systems such as single-degree or multi-degree with non-linearities were taken up and corresponding equivalent linear systems were found using the optimization principles. This task was carried out under free or forced vibration conditions. Different types of stiffness as well as damping non-linearities were also investigated. The results thus obtained were compared with those obtained by other researchers. In this way, the first contribution of this thesis is the use of the optimization principles in finding equivalent linear systems.

In Chapter 3, the dynamic equations of motion of the rotor system were derived using the Lagrange equation as well as the influence coefficient method. The time domain solution of the non-linear set of equations were obtained using the Newmark-β Method in conjunction with an iteration procedure. The equivalent linear system was found using the optimization principles discussed in Chapter 2. The balance weights for the non-linear system were obtained by balancing the equivalent linear system using least squares influence coefficient method. The balancing results clearly showed the effectiveness of this method.

4.2 Recommendations for Future Work

There are some interesting studies which should be attempted in the future. These include the following:

1. The effect of the variation of non-linear stiffness values of the bearings on the balance weights.
2. The effect of the balancing plane location on the rotor response and balancing weights.
3. Inclusion of the effects of other non-linear rotor parameters such as non-linear damping and seal forces in the rotor balancing model.

REFERENCES

Bandstra, J.P. (1983). "Comparison of Equivalent Viscous Damping and Nonlinear Damping in Discrete and Continuous Vibrating Systems", Journal of Vibration, Acoustics, Stress, And Reliability in Design, Vol.105, pp. 382-392.

Bishop, R.E.D. (1959). "The Vibrations of Rotating Shafts," Journal of Mechanical Engineering Science, Vol.1, No.1, pp.50-64.

Bishop, R.E.D., and Parkinson, A.G. (1972). "On the Use of Balancing Machines for Flexible Rotors", Transactions of the American Society of Mechanical Engineers, Journal of Engineering for Industry, Vol.94, pp.561-576.

Bishop, R.E.D., and Gladwell, G.M.L. (1959). "The Vibration and Balancing of an Unbalanced Flexible Rotor," Journal of Mechanical Engineering Science, Vol.1, No.1, pp.66-77.

Darlow, M.S. (1989). Balancing of High-Spec. Machinery, Springer-Verlag New York Inc., New York, NY, USA, pp 67-79.

Genta, G., and Repaci, A. (1988). "Circular Whirling and Unbalance Response of Nonlinear Rotors, Rotating Machinery Dynamics, Vol 2, Bentley Rotor Dynamics Research Corporation, 1988, pp. 441-448.

Géradin, M., and ''ill, N. (1988). "Non-linear Dynamic Analysis of Flexible Rotors", Proceedings of Institution of Mechanical Engineering International Conference on Vibrations in Rotating Machinery, Sept. 1988, pp. 627-634.

Gladwell, G.M.L., and Bishop, R.E.D. (1959). "The Vibration and Balancing of Rotating Shafts Supported in Flexible Bearings," Journal of Mechanical Science, Vol 1, No.3, p.195.

Goodman, T.P. (1964). "A Least Square Method for Computing Balance Corrections", Transactions of the American Society of Mechanical Engineers, Journal of Engineering For Industry, Vol.86, No.3, pp. 273-279.

Guang, M., Jian-Kang, X., and Zhong-Qing, X. (1988). "Application of Equivalent Linearizing Method to the Analyses of Flexible Rotor - Uncentralized Squeeze Film Damper Bearing System", Rotating Machinery Dynamics, Vol 2, Bentley Rotor Dynamics Research Corporation, 1988, pp. 115-119.

Jones, S.E. (1978). "Remarks on the Perturbation Method For Certain Conservative Systems", Journal of Nonlinear Mechanics, Vol.13, pp.125-128.

Kellenberger, W. (1972). "Should a Flexible Rotor Be Balanced in N or (N + 2) Planes", Transactions of the American Society of mechanical Engineers, Journal of Engineering For Industry, Vol.94, pp. 548-560.

Krylov, N. and Bogoliubov, N.N., (1943), Introduction To Nonlinear Mechanics, Princeton University Press, Princeton, N.J., pp. 8-28.

Little, R.M., and Pilkey, W.D. (1976). "A Linear Programming Approach for Balancing Flexible Rotors", Transactions of the American Society of Mechanical Engineers, Journal of Engineering for Industry, Vol.98, No.3, pp. 1030-1035.

Lund, J.W., and Tonneson, J. (1972). "Analysis And Experiments in Multiplane Balancing of Flexible Rotors", Transactions of the American Society of Mechanical Engineers, Journal of Engineering For Industry, Vol.94, No.1, pp. 233-242.

Mehlahl, A. (1954). "Auswuchten Elastischer Rotoren", Z.angew. Match und Mech., Vol.34.

Nayfeh, A.H., and Mook, D.T., (1979), Nonlinear Oscillations, Wiley - Interscience Series of Texts, Monographs & Tracts, pp.617-686.

Rakheja, S., Van Valiet, M., and Sankar, S. (1985). "A Discrete Harmonic Linearization Technique For Simulating Non-Linear Mechanical Systems", Journal of Sound Vibrations, Vol. 100, No.4, June 22, 1985, pp. 511-526.

Rao, A.C. (1985). "Energy Correlation Concept for the Solution of Non-Linear Forced Vibration Problems", Indian Journal of Technology, Vol 23, August (1985), pp. 299-301.

Rao, J.S. (1983), Rotor Dynamics, Wiley Eastern Ltd., New Delhi, India, pp. 207-214.

Rao, S.S. (1978). Optimization Theory and Applications, Wiley Eastern Limited, New Delhi, India, pp. 317-327.

Rao, S.S. (1986-a). Mechanical Vibrations, Addison-Wesley Publishing Company, Inc., Massachusetts, USA, p. 185.

Rao, S.S. (1986-b). Mechanical Vibrations, Addison-Wesley Publishing Company, Inc., Massachusetts, USA, pp. 489-491.

Rieger, N.F. (1967). "Computer Program for Balancing of Flexible Rotors", Mechanical Technology Incorporated Report 67TR68, September (1967).

Sharan, A.M., and Rao, J.S. (1985). "Unbalance Response of Rotor Disks Supported by Fluid Film Bearings with a Negative Cross Coupled Stiffness Using Infleunce Coefficient Method", Mechanism and Meachine Theory, Vol. 20, No. 5, pp. 415-426.

Sharan, A.M., and Turpin, A.E. (1989). "Solution of Non-Linear Vibration Problems Using Optimization Principles", 12th Biennial Conference on Mechanical Vibration and Noise, Montreal, Quebec, Canada, Vol.18-5.

Sinha, S.C. and Srinivasan, P. (1971-a). "Application of Ultraspherical Polynomials to Nonlinear Autonomous Systems", Journal of Sound And Vibration, Vol.18(1), pp. 55-60.

Sinha, S.C. and Srinivasan, P. (1971-b). "A Weighted Mean Square Method of Linearization In Nonlinear Oscillations", Journal of Sound and Vibration, Vol.16(2), pp. 139-148.

Stoker, J.J., (1950), Nonlinear Vibrations, Interscience Publishers Inc., New York, pp. 90-91.

Tessarzik, J.M., and Badgley, R.H. (1974). "Experimental Evaluation of The Exact Point-Speed and Least Squares Procedures For Flexible Rotor Balancing by the Influence Coefficient Method", Transaction of the American Society of Mechanical Engineers, Journal of Engineering for Industry, Vol.96, No.2, pp. 633-643.

Tessarzik, J.M., Badgley R.H., and Anderson, W.J. (1972). "Flexible Rotor Balancing by the Exact Point-Speed Influence Coefficient Method", Transactions of the American Society of Mechanical Engineers, Journal of Engineering for Industry, Vol.94, No.1, pp. 148-158.

Tonneson, J. (1974). "Further Experiments on Balancing of a High Speed Flexible Rotor", Transactions of the American Society of Mechanical Engineers, Journal of Engineering for Industry, Vol.96, No.2, pp. 431-440.

Tse, F.S., Morse, I.E., and Hinkle, R.T. (1978-a). Mechanical Vibrations, Theory and Applications, Allyn and Bacon, Inc., Boston, Massachusetts, USA, pp. 406-407.

Tse, F.S., Morse, I.E., and Hinkle, R.T. (1978). Mechanical Vibrations, Theory and Applications, Allyn and Bacon, Inc., Boston, Massachusetts, USA, pp.356-360.

APPENDICES

APPENDIX A

MODAL ANALYSIS - MDOF SYSTEM

The linear differential equations for the free vibration of the multi-degree-of-freedom system in Case No. 5, Section 2.2.3.5 is given by:

$$[M]\ddot{x}(t) + [C]\dot{x}(t) + [K]x(t) = \{0\} \tag{A.1}$$

The three second order differential equations can be expressed as six first order differential equations:

$$[A]\{\dot{q}(t)\} + [B]\{q(t)\} = \{0\} \tag{A.2}$$

where

$$[A] = \begin{bmatrix} [0] & [M] \\ [M] & [C] \end{bmatrix} \tag{A.3}$$

$$[B] = \begin{bmatrix} -[M] & [0] \\ [0] & [K] \end{bmatrix} \tag{A.4}$$

$$\{q(t)\} = \begin{Bmatrix} \dot{x}(t) \\ x(t) \end{Bmatrix} \tag{A.5}$$

Assuming the eigensolution in the form

$$q(t) = e^{\alpha_j(t)}\{\Psi_j\} \qquad j=1\cdots6 \tag{A.6}$$

where α_j = jth complex eigenvalue,

Ψ_j = jth complex eigenvector corresponding to α_j,

from Eqs. (A.2) and (A.6) we have

$$[g(\alpha)]\ \{\Psi\} = \{0\} \tag{A.7}$$

where $$[g(\alpha)] = [D] - \frac{1}{\alpha}[I_1] \tag{A.8}$$

Here [D] is the dynamic matrix, and $[I_1]$ is an identity matrix. The matrix [D] is found to be:

$$[D] = [-B]^{-1}[A] \tag{A.9}$$

$$[D] = \begin{bmatrix} [0] & [I_2] \\ -[K]^{-1}[M] & -[K]^{-1}[C] \end{bmatrix} \tag{A.10}$$

From Eqs. (A.8) and (A.7) the complex eigenvalues and corresponding complex eigenvectors for the reduced first order equations can be found.

Introducing a new state vector {Z}, defined as

$$\{q(t)\} = [\Psi]\{Z(t)\} \tag{A.11}$$

which can be written as:

$$\{Z(t)\} = [\Psi]^{-1}\{q(t)\} \tag{A.12}$$

Substituting Eq. (A.12) into Eq. (A.2) and premultiplying by $[\Psi]^T$ we obtain:

$$[A^*]\{\dot{Z}(t)\} + [B^*]\{Z(t)\} = \{0\} \tag{A.13}$$

where $[A^*]$ and $[B^*]$ are diagonal matrices

$$[A^*] = [\Psi]^T[A][\Psi] \tag{A.14}$$

$$[B^*] = [\Psi]^T[B][\Psi] \tag{A.15}$$

The initial conditions in the {Z} coordinates are found from Eq. (A.12) such that:

$$\{Z(0)\} = \{\Psi\}^{-1}\{q(0)\} = \{\Psi\}^{-1}\begin{Bmatrix}\dot{x}(0)\\x(0)\end{Bmatrix} \tag{A.16}$$

Let Z_{i0} be the initial condition for the i^{th} mode, the solution of Eq. (A.13) is of the form

$$z_i = z_{io}e^{\alpha_i t} \tag{A.17}$$

Expressing the complex eigenvalue in the general form

$$\alpha_i = -\zeta_i + j\omega_i \tag{A.18}$$

the solution of the i^{th} mode can be written as

$$z_i(t) = z_{io}e^{-\zeta_i t}\{\cos(\omega_i t) + j\sin(\omega_i t)\} \qquad i=1\cdots6 \tag{A.19}$$

Finally the solution in {x} coordinates is determined by

$$\{q(t)\} = \begin{Bmatrix}\dot{x}(t)\\x(t)\end{Bmatrix} = [\Psi]\{z(t)\} \tag{A.20}$$

APPENDIX B

DERIVATION OF SHAFT STIFFNESS MATRIX [K_s]

Consider a supported beam as shown in Fig. B.1(a). In general the deflection due to load W is given by:

$$u(x) = \frac{1}{EI}\left[\frac{W(l-b)}{6l}x^3 - \frac{W(x-b)^3}{6} + \frac{(W(l-b)x)*((l-b)^2 l-l^3)}{6l^2}\right] \tag{B.1}$$

Using Eq. (B.1) the flexibility influence coefficients $a_{2,j}$,(j=1,4) for the rotor in Fig. B.1(b) can be found by applying a unit load at the location for m_2, similarly $a_{3,j}$,(j=1,4) can be found by applying a unit load at the location of m_3 and so on. From Maxwell's reciprocity theorem $a_{i,j} = a_{j,i}$. The coefficients are as follows:

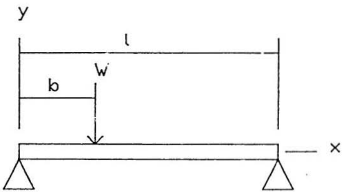
$$a_{22} = \frac{1}{EI}\left(\frac{(L-l_2)(l_2)^3}{6L} + \frac{(L-l_2)(l_2)((L-l_2)^2L-L^3)}{6L^2}\right)$$

$$a_{32} = \frac{1}{EI}\left(\frac{(L-l_2)(l_3)^3}{6L} - \frac{(l_3-l_2)^3}{6} + \frac{(L-l_2)(l_3)((L-l_2)^2L-L^3)}{6L^2}\right)$$

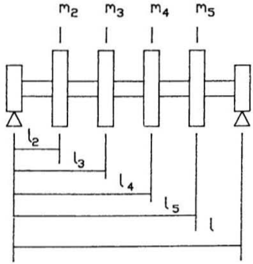
$$a_{42} = \frac{1}{EI}\left(\frac{(L-l_2)(l_4)^3}{6L} - \frac{(l_4-l_2)^3}{6} + \frac{(L-l_2)(l_4)((L-l_2)^2L-L^3)}{6L^2}\right)$$

$$a_{52} = \frac{1}{EI}\left(\frac{(L-l_2)(l_5)^3}{6L} - \frac{(l_5-l_2)^3}{6} + \frac{(L-l_2)(l_5)((L-l_2)^2L-L^3)}{6L^2}\right)$$

$$a_{23} = a_{32}$$



a) Beam



b) Rotor

FIG. B1

$$a_{33} = \frac{1}{EI}\left(\frac{(L-l_3)(l_3)^3}{6L} + \frac{(L-l_3)(l_3)\{(L-l_3)^2L-L^3\}}{6L^2}\right)$$

$$a_{43} = \frac{1}{EI}\left(\frac{(L-l_3)(l_4)^3}{6L} - \frac{(l_4-l_3)^3}{6} + \frac{(L-l_3)(l_4)\{(L-l_3)^2L-L^3\}}{6L^2}\right)$$

$$a_{53} = \frac{1}{EI}\left(\frac{(L-l_3)(l_5)^3}{6L} - \frac{(l_5-l_3)^3}{6} + \frac{(L-l_3)(l_5)\{(L-l_3)^2L-L^3\}}{6L^2}\right)$$

$$a_{24} = a_{42}$$

$$a_{34} = a_{43}$$

$$a_{44} = \frac{1}{EI}\left(\frac{(L-l_4)(l_4)^3}{6L} + \frac{(L-l_4)(l_4)\{(L-l_4)^2L-L^3\}}{6L^2}\right)$$

$$a_{54} = \frac{1}{EI}\left(\frac{(L-l_4)(l_5)^3}{6L} - \frac{(l_5-l_4)^3}{6} + \frac{(L-l_4)(l_5)\{(L-l_4)^2L-L^3\}}{6L^2}\right)$$

$$a_{25} = a_{52}$$

$$a_{35} = a_{53}$$

$$a_{45} = a_{54}$$

$$a_{55} = \frac{1}{EI}\left(\frac{(L-l_5)(l_5)^3}{6L} + \frac{(L-l_5)(l_5)\{(L-l_5)^2L-L^3\}}{6L^2}\right)$$

The flexibility matrix for the shaft is given by:

$$[a] = \begin{bmatrix} a_{22} & a_{23} & a_{24} & a_{25} \\ a_{32} & a_{33} & a_{34} & a_{35} \\ a_{42} & a_{43} & a_{44} & a_{45} \\ a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix}$$

The shaft stiffness matrix can be found from the relationship:

$$[K_s] = [a]^{-1}$$

$$[K_s] = \begin{bmatrix} k_{22} & k_{23} & k_{24} & k_{25} \\ k_{32} & k_{33} & k_{34} & k_{35} \\ k_{42} & k_{43} & k_{44} & k_{45} \\ k_{52} & k_{53} & k_{54} & k_{55} \end{bmatrix}$$

APPENDIX C

DERIVATION OF THE EQUATIONS OF MOTION FOR A ROTOR SUPPORTED ON NONLINEAR BEARINGS

As discussed in Chapter 3, the equations of motion for a multi-mass rotor shown in Fig. 3.2, are formulated using Lagrange Equations and the influence coefficient method. Using the Lagrange approach the system of equations are derived from two scalar functions, kinetic energy, and potential energy. Applying this method, the equations of motion for the rotor example given in Fig. 3.4 are found.

From Eqs. (3.1) and (3.5), and using the shaft stiffness matrix $[K_s]$ from Appendix B, the total potential energy for the rotor, V_T, is

$$V_T = \int_0^{z_1} (K_{zz}^{1B} z + \varepsilon^{1B} z^3)\, dz + \int_0^{y_1} (K_{yy}^{1B} y + \varepsilon^{1B} y^3)\, dy + \int_0^{z_6} (K_{zz}^{2B} z + \varepsilon^{2B} z^3)\, dz$$

$$+ \int_0^{y_6} (K_{yy}^{2B} y + \varepsilon^{2B} y^3)\, dy + \frac{1}{2} [\, (z_2 - z_2^{\bullet})\ (z_3 - z_3^{\bullet})\ (z_4 - z_4^{\bullet})\ (z_5 - z_5^{\bullet})\,]$$

$$\times \begin{bmatrix} k_{22} & k_{23} & k_{24} & k_{25} \\ k_{32} & k_{33} & k_{34} & k_{35} \\ k_{42} & k_{43} & k_{44} & k_{45} \\ k_{52} & k_{53} & k_{54} & k_{55} \end{bmatrix} \begin{bmatrix} z_2 - z_2^{\bullet} \\ z_3 - z_3^{\bullet} \\ z_4 - z_4^{\bullet} \\ z_5 - z_5^{\bullet} \end{bmatrix}$$

$$+ \frac{1}{2} [\, (y_2 - y_2^{\bullet})\ (y_3 - y_3^{\bullet})\ (y_4 - y_4^{\bullet})\ (y_5 - y_5^{\bullet})\,]$$

$$\times \begin{bmatrix} k_{22} & k_{23} & k_{24} & k_{25} \\ k_{32} & k_{33} & k_{34} & k_{35} \\ k_{42} & k_{43} & k_{44} & k_{45} \\ k_{52} & k_{53} & k_{54} & k_{55} \end{bmatrix} \begin{bmatrix} y_2 - y_2^{\bullet} \\ y_3 - y_3^{\bullet} \\ y_4 - y_4^{\bullet} \\ y_5 - y_5^{\bullet} \end{bmatrix} \tag{C.1}$$

Expanding Eq. (C.1) and using the relationship $k_{ij} = k_{ji}$, the expression for the total potential energy becomes

$$
\begin{aligned}
V_T = \; & \int_0^{z_1}(K_{zz}^{1B}z+\varepsilon^{1B}z^3)dz+\int_0^{y_1}(K_{yy}^{1B}y+\varepsilon^{1B}y^3)dy \\
& +\int_0^{z_6}(K_{zz}^{2B}z+\varepsilon^{2B}z^3)dz + \int_0^{y_6}(K_{yy}^{2B}y+\varepsilon^{2B}y^3)dy \\
& +\frac{1}{2}k_{22}(z_2-z_2^*)^2+k_{23}(z_3-z_3^*)(z_2-z_2^*)+k_{24}(z_4-z_4^*)(z_2-z_2^*) \\
& +k_{25}(z_5-z_5^*)(z_2-z_2^*)+\frac{1}{2}k_{33}(z_3-z_3^*)^2+k_{34}(z_4-z_4^*)(z_3-z_3^*) \\
& +k_{35}(z_5-z_5^*)(z_3-z_3^*)+\frac{1}{2}k_{44}(z_4-z_4^*)^2+k_{45}(z_5-z_5^*)(z_4-z_4^*)+\frac{1}{2}k_{55}(z_5-z_5^*)^2 \\
& +\frac{1}{2}k_{22}(y_2-y_2^*)^2+k_{23}(y_3-y_3^*)(y_2-y_2^*)+k_{24}(y_4-y_4^*)(y_2-y_2^*) \\
& +k_{25}(y_5-y_5^*)(y_2-y_2^*)+\frac{1}{2}k_{33}(y_3-y_3^*)^2+k_{34}(y_4-y_4^*)(y_3-y_3^*) \\
& +k_{35}(y_5-y_5^*)(y_3-y_3^*)+\frac{1}{2}k_{44}(y_4-y_4^*)+k_{45}(y_5-y_5^*)(y_4-y_4^*)+k_{55}(y_5-y_5^*)^2
\end{aligned}
\tag{C.2}
$$

From Eq. (3.6) the kinetic energy T, for the rotor in Fig. 3.3 is given as

$$
\begin{aligned}
T = \; & \frac{1}{2}m_1\left(\frac{dz_1}{dt}\right)^2+\frac{1}{2}m_2\left(\frac{d}{dt}(z_2+r_2\cos(\omega t+\phi_2))\right)^2 \\
& +\frac{1}{2}m_3\left(\frac{d}{dt}(z_3+r_3\cos(\omega t+\phi_3))\right)^2+\frac{1}{2}m_4\left(\frac{d}{dt}(z_4+r_4\cos(\omega t+\phi_4))\right)^2 \\
& +\frac{1}{2}m_5\left(\frac{d}{dt}(z_5+r_5\cos(\omega t+\phi_5))\right)^2+\frac{1}{2}m_6\left(\frac{dz_6}{dt}\right)^2 \\
& +\frac{1}{2}m_1\left(\frac{dy_1}{dt}\right)^2+\frac{1}{2}m_2\left(\frac{d}{dt}(y_2+r_2\sin(\omega t+\phi_2))\right)^2 \\
& +\frac{1}{2}m_3\left(\frac{d}{dt}(y_3+r_3\sin(\omega t+\phi_3))\right)^2+\frac{1}{2}m_4\left(\frac{d}{dt}(y_4+r_4\sin(\omega t+\phi_4))\right)^2 \\
& +\frac{1}{2}m_5\left(\frac{d}{dt}(y_5+r_5\sin(\omega t+\phi_5))\right)^2+\frac{1}{2}m_6\left(\frac{dy_6}{dt}\right)^2
\end{aligned}
\tag{C.3}
$$

Differentiating we get

$$
\begin{aligned}
T = & \frac{1}{2}m_1(\dot{z}_1)^2 \\
& +\frac{1}{2}m_2(\dot{z}_2-r_2\omega\sin(\omega t+\phi_2))^2 \\
& +\frac{1}{2}m_3(\dot{z}_3-r_3\omega\sin(\omega t+\phi_3))^2 \\
& +\frac{1}{2}m_4(\dot{z}_4-r_4 w\sin(\omega t+\phi_4))^2 \\
& +\frac{1}{2}m_5(\dot{z}_5-r_5\omega\sin(\omega t+\phi_5))^2 \\
& +\frac{1}{2}m_6(\dot{z}_6)^2 \\
& +\frac{1}{2}m_1(\dot{y}_1)^2 \\
& +\frac{1}{2}m_2(\dot{y}_2+r_2\omega\cos(\omega t+\phi_2))^2 \\
& +\frac{1}{2}m_3(\dot{y}_3+r_3\omega\cos(\omega t+\phi_3))^2 \\
& +\frac{1}{2}m_4(\dot{y}_4+r_4\omega\cos(\omega t+\phi_4))^2 \\
& +\frac{1}{2}m_5(\dot{y}_5+r_5\omega\cos(\omega t+\phi_5))2 \\
& +\frac{1}{2}m_6(\dot{y}_6)^2
\end{aligned}
\tag{C.4}
$$

Having found the kinetic and potential energy expressions, we can introduce the Lagrangian, defined by:

$$L^{*} = T - V_T \tag{C.5}$$

and the Lagrange's equations of motion, which can be written as

$$\frac{d}{dt}\left(\frac{\partial L^{*}}{\partial \dot{q}_i}\right) - \frac{\partial L^{*}}{\partial q_i}+\frac{\partial \psi}{\partial q_i} = Q_i \qquad i = 1...n \tag{C.6}$$

In Eq. (C.6) Q_i represents the nonconservative forces, q_i the generalized coordinate, and ψ represents the Rayleigh's dissipation function. In this rotor example, there is no viscous damping and therefore $d\psi/d\dot{q}_i = 0$.

Substituting Eq. (C.5) into Eq. (C.6), and eliminating zero terms, Eq. (C.6) becomes

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) + \frac{\partial V_T}{\partial q_i} = 0 \tag{C.7}$$

The rigid body mode displacements z_i^* and y_i^* can be eliminated from the expressions for kinetic and potential energy by using Eq. (3.9):

$$\begin{aligned} z_2^* &= z_1 + l_2 \tan\theta = z_1 + l_2\left(\frac{z_6 - z_1}{L}\right) = z_1\left(1 - \frac{l_2}{L}\right) + \frac{l_2}{L} z_6 \\ &= a_1 z_1 + a_2 z_2 \\ &\textit{where}\quad a_1 = \left(1 - \frac{l_2}{L}\right) \qquad a_2 = \left(\frac{l_2}{L}\right) \end{aligned} \tag{C.8}$$

$$\begin{aligned} z_3^* &= z_1 + l_3 \tan\theta = z_1 + l_3\left(\frac{z_6 - z_1}{L}\right) = z_1\left(1 - \frac{l_3}{L}\right) + \frac{l_3}{L} z_6 \\ &= a_3 z_1 + a_4 z_6 \\ &\textit{where}\quad a_3 = \left(1 - \frac{l_3}{L}\right) \qquad a_4 = \left(\frac{l_3}{L}\right) \end{aligned} \tag{C.9}$$

$$\begin{aligned} z_4^* &= z_1 + l_4 \tan\theta = z_1 + l_4\left(\frac{z_6 - z_1}{L}\right) = z_1\left(1 - \frac{l_4}{L}\right) + \frac{l_4}{L} z_6 \\ &= a_5 z_1 + a_6 z_6 \\ &\textit{where}\quad a_5 = \left(1 - \frac{l_4}{L}\right) \qquad a_6 = \left(\frac{l_4}{L}\right) \end{aligned} \tag{C.10}$$

$$z_5^* = z_1 + l_5 \tan\theta = z_1 + l_5\left(\frac{z_6 - z_1}{L}\right) = z_1\left(1 - \frac{l_5}{L}\right) + \frac{l_5}{L}$$
$$= a_7 z_1 + a_8 z_6$$
$$\text{where} \quad a_7 = \left(1 - \frac{l_5}{L}\right) \qquad a_8 = \left(\frac{l_5}{L}\right) \tag{C.11}$$

$$\text{and where} \qquad \theta = \frac{z_6 - z_1}{L} \tag{C.12}$$

similar expressions can be obtained for y_i^* using Eq. (3.9).

Using Eq. (C.7) and Eqs. (C.8) to (C.12) the equations of motion for the rotor are found as:

$$[M]\begin{Bmatrix} \ddot{z}_1 \\ \ddot{y}_1 \\ \ddot{z}_2 \\ \ddot{y}_2 \\ \cdot \\ \cdot \\ \ddot{z}_6 \\ \ddot{y}_6 \end{Bmatrix} + [K]\begin{Bmatrix} z_1 \\ y_1 \\ z_2 \\ y_2 \\ \cdot \\ \cdot \\ z_6 \\ y_6 \end{Bmatrix} = \{F\} \tag{C.13}$$

where [M] is a diagonal mass matrix, [K] is the overall stiffness matrix, and [F] is the force vector. Structural damping was then added to the rotor model. Introducing a diagonal modal damping matrix $[\bar{C}]$:

$$[\bar{C}] = 2\begin{bmatrix} \zeta_1\omega_1 m_{1,1} & 0 & \cdots & 0 \\ 0 & \zeta_2\omega_2 m_{2,2} & \cdots & \cdots \\ & \ddots & & \\ & & \ddots & \\ & & & \zeta_{12}\omega_{12} m_{12,12} \end{bmatrix}$$

where in this case, ω_i were the eigenvalues for the rotor system excluding the non-linear bearing stiffness terms, ζ_i were assumed modal damping coefficients, and $m_{i,j}$ were the terms from the mass matrix [M]. The full structural damping matrix [C] was found from the following expression:

$$[C] = [\Phi^T]^{-1}[\bar{C}][\Phi]^{-1} \tag{C.15}$$

where $[\Phi]$ is the matrix of eigenvectors, again, for the rotor excluding the non-linear bearing stiffness terms. The equations of motion for the rotor are as follows:

$$[M]\begin{Bmatrix} \ddot{z}_1 \\ \ddot{y}_1 \\ \ddot{z}_2 \\ \ddot{y}_2 \\ \vdots \\ \ddot{z}_6 \\ \ddot{y}_6 \end{Bmatrix} + [C]\begin{Bmatrix} \dot{z}_1 \\ \dot{y}_1 \\ \dot{z}_2 \\ \dot{y}_2 \\ \vdots \\ \dot{z}_6 \\ \dot{y}_6 \end{Bmatrix} + [K]\begin{Bmatrix} z_1 \\ y_1 \\ z_2 \\ y_2 \\ \vdots \\ z_6 \\ y_6 \end{Bmatrix} = \{F\} \tag{C.16}$$

The elements of matrices [M] and [K], and force vector {F} in Eq. (C.16) are as follows.

Elements of Mass Matrix [M]

[M] is a diagonal matrix with the following diagonal elements:

$M_{11} = m_1$
$M_{22} = m_1$
$M_{33} = m_2$
$M_{44} = m_2$
$M_{55} = m_3$
$M_{66} = m_3$
$M_{77} = m_4$
$M_{88} = m_4$
$M_{99} = m_5$
$M_{1010} = m_5$
$M_{1111} = m_6$
$M_{1212} = m_6$

Elements of the Stiffness Matrix [K]

$$
\begin{aligned}
K_{11} = {} & K_{zz}^{1B} + (a_2^2 * k_{22}) + (2 * k_{23} * a_1 * a_3) + (2 * k_{24} * a_1 * a_5) \\
& + (2 * k_{25} * a_1 * a_7) + (a_3^2 * k_{33}) + (2 * k_{34} * a_3 * a_5) \\
& + (2 * k_{35} * a_3 * a_7) + (a_5^2 * k_{44}) + (2 * k_{45} * a_5 * a_7) \\
& + (a_7^2 * k_{55}) + \varepsilon^{1B} z_1^2
\end{aligned}
$$

$K_{12} = 0.0$
$K_{13} = (-a_1 * k_{22}) - (a_3 * k_{23}) - (a_5 * k_{24}) - (a_7 * k_{25})$
$K_{14} = 0.0$
$K_{15} = (-a_1 * k_{23}) - (a_3 * k_{33}) - (a_5 * k_{34}) - (a_7 * k_{35})$
$K_{16} = 0.0$
$K_{17} = (-a_1 * k_{24}) - (a_3 * k_{34}) - (a_5 * k_{44}) - (a_7 * k_{45})$
$K_{18} = 0.0$
$K_{19} = (-a_1 * k_{25}) - (a_3 * k_{35}) - (a_5 * k_{45}) - (a_7 * k_{55})$
$K_{1\ 10} = 0.0$

$$
\begin{aligned}
K_{1\ 11} = {} & (a_1 * a_2 * k_{22}) + (a_1 * a_4 * k_{23}) + (a_2 * a_3 * k_{23}) + (a_2 * a_5 * k_{24}) \\
& + (a_1 * a_6 * k_{24}) + (a_2 * a_7 * k_{25}) + (a_1 * a_8 * k_{25}) + (a_3 * a_4 * k_{33}) \\
& + (a_3 * a_6 * k_{34}) + (a_4 * a_5 * k_{35}) + (a_4 * a_7 * k_{35}) + (a_3 * a_8 * k_{35}) \\
& + (a_5 * a_6 * k_{44}) + (a_6 * a_7 * k_{45}) + (a_5 * a_8 * k_{45}) + (a_7 * a_8 * k_{55})
\end{aligned}
$$

$K_{1\ 12} = 0.0$

$K_{21}=0.0$

$K_{22}=K_{yy}^{1B}+(a_1^2*k_{22})+(2*k_{23}*a_1*a_3)+(2*k_{24}*a_1*a_5)$
$+(2*k_{25}*a_1*a_7)+(2*k_{34}*a_3*a_5)+(2*k_{35}*a_3*a_7)+$
$(a_5^2*k_{44})+(2*k_{45}*a_5*a_7)+(a_7^2*k_{55})+(a_3^2*k_{33})+\varepsilon^{1B}y_1^2$

$K_{23}=0.0$

$K_{24}=(-a_1*k_{22})-(a_3*k_{23})-(a_5*k_{24})-(a_7*k_{25})$

$K_{25}=0.0$

$K_{26}=(-a_1*k_{23})-(a_3*k_{33})-(a_5*k_{34})-(a_7*K_{35})$

$K_{27}=0.0$

$K_{28}=(-a_1*K_{24})-(a_3*K_{34})-(a_5*K_{44})-(a_7*k_{45})$

$K_{29}=0.0$

$K_{2\ 10}=(-a_1*k_{25})-(a_3*k_{35})-(a_5*k_{45})-(a_7*k_{55})$

$K_{2\ 11}=0.0$

$K_{2\ 12}=(a_1*a_2*k_{22})+(a_1*a_4*k_{23})+(a_2*a_3*k_{23})+(a_2*a_5*k_{24})+$
$(a_1*a_6*k_{24})+(a_2*a_7*k_{25})+(a_1*a_8*k_{25})+(a_3*a_4*k_{33})+$
$(a_3*a_6*k_{34})+(a_4*a_5*k_{34})+(a_4*a_7*k_{35})+(a_3*a_8*k_{35})+$
$(a_5*a_6*k_{44})+(a_6*a_7*K_{45})+(a_5*a_8*K_{45})+(a_7*a_8*K_{55})$

$K_{31}=(-a_1*k_{22})-(a_3*k_{23})-(a_5*k_{24})-(a_7*k_{25})$

$K_{32}=0.0$

$K_{33}=k_{22}$

$K_{34}=0.0$

$K_{35}=k_{23}$

$K_{36}=0.0$

$K_{37}=k_{24}$

$K_{38}=0.0$

$K_{39}=k_{25}$

$K_{3\ 10}=0.0$

$K_{3\ 11}=(-a_2*k_{22})-(a_4*k_{23})-(a_6*k_{24})-(a_8*K_{25})$

$K_{3\ 12}=0.0$

$K_{41}=0.0$
$K_{42}=(-a_1*k_{22})-(a_3*k_{23})-(a_5*k_{24})-(a_7*k_{25})$
$K_{43}=0.0$
$K_{44}=k_{22}$
$K_{45}=0.0$
$K_{46}=k_{23}$
$K_{47}=0.0$
$K_{48}=k_{24}$
$K_{49}=0.0$
$K_{4\ 10}=k_{25}$
$K_{4\ 11}=0.0$
$K_{4\ 12}=(-a_2*k_{22})-(a_4*k_{23})-(a_6*k_{24})-(a_8*k_{25})$

$K_{51}=(-a_1*k_{23})-(a_3*k_{33})-(a_5*k_{34})-(a_7*k_{35})$
$K_{52}=0.0$
$K_{53}=k_{23}$
$K_{54}=0.0$
$K_{55}=k_{33}$
$K_{56}=0.0$
$K_{57}=k_{34}$
$K_{58}=0.0$
$K_{59}=k_{35}$
$K_{5\ 10}=0.0$
$K_{5\ 11}=(-a_2*k_{23})-(a_4*k_{33})-(a_6*k_{34})-(a_8*K_{35})$
$K_{5\ 12}=0.0$
$K_{61}=0.0$
$K_{62}=(-a_1*k_{23})-(a_3*k_{33})-(a_5*k_{34})-(a_7*k_{35})$
$K_{63}=0.0$
$K_{64}=k_{23}$
$K_{65}=0.0$
$K_{66}=k_{33}$
$K_{67}=0.0$
$K_{68}=k_{34}$
$K_{69}=0.0$
$K_{6\ 10}=k_{35}$
$K_{6\ 11}=0.0$
$K_{6\ 12}=(-a_2*k_{23})-(a_4*k_{33})-(a_6*k_{34})-(a_8*K_{35})$

$K_{71}=(-a_1*k_{24})-(a_3*k_{34})-(a_5*k_{44})-(a_7*k_{45})$
$K_{72}=0.0$
$K_{73}=k_{24}$
$K_{74}=0.0$
$K_{75}=k_{34}$
$K_{76}=0.0$
$K_{77}=k_{44}$
$K_{78}=0.0$
$K_{79}=k_{45}$
$K_{7\ 10}=0.0$
$K_{7\ 11}=(-a_2*k_{24})-(a_4*k_{34})-(a_6*k_{44})-(a_8*k_{45})$
$K_{7\ 12}=0.0$

$K_{81}=0.0$
$K_{82}=(-a_1*k_{24})-(a_3*k_{34})-(a_5*k_{44})-(a_7*k_{45})$
$K_{83}=0.0$
$K_{84}=k_{24}$
$K_{85}=0.0$
$K_{86}=k_{34}$
$K_{87}=0.0$
$K_{88}=k_{44}$
$K_{89}=0.0$
$K_{8\ 10}=k_{45}$
$K_{8\ 11}=0.0$
$K_{8\ 12}=(-a_2*k_{24})-(a_4*k_{34})-(a_6*k_{44})-(a_8*k_{45})$

$K_{91}=(-a_1*k_{25})-(a_3*k_{35})-(a_5*k_{45})-(a_7*k_{55})$
$K_{92}=0.0$
$K_{93}=k_{25}$
$K_{94}=0.0$
$K_{95}=k_{35}$
$K_{96}=0.0$
$K_{97}=k_{45}$
$K_{98}=0.0$
$K_{99}=k_{55}$
$K_{9\ 10}=0.0$
$K_{9\ 11}=(-a_2*k_{25})-(a_4*k_{35})-(a_6*k_{45})-(a_8*k_{55})$
$K_{9\ 12}=0.0$

$K_{10\ 1}=0.0$
$K_{10\ 2}=(-a_1*k_{25})-(a_3*k_{35})-(a_5*k_{45})-(a_7*k_{55})$
$K_{10\ 3}=0.0$
$K_{10\ 4}=k_{25}$
$K_{10\ 5}=0.0$
$K_{10\ 6}=k_{35}$
$K_{10\ 7}=0.0$
$K_{10\ 8}=k_{45}$
$K_{10\ 9}=0.0$
$K_{10\ 10}=k_{55}$
$K_{10\ 11}=0.0$
$K_{10\ 12}=(-a_2*k_{25})-(a_4*k_{35})-(a_6*k_{45})-(a_8*k_{55})$

$$
\begin{aligned}
K_{11\ 1}=&(a_1*a_2*k_{22})+(a_2*a_3*k_{23})+(a_1*a_4*k_{23})+(a_2*a_5*k_{24})+\\
&(a_1*a_6*k_{24})+(a_2*a_7*k_{25})+(a_1*a_8*k_{25})+(a_3*a_4*k_{33})+\\
&(a_3*a_6*k_{34})+(a_4*a_5*k_{34})+(a_4*a_7*k_{35})+(a_3*a_8*k_{35})+\\
&(a_5*a_6*k_{44})+(a_6*a_7*k_{45})+(a_5*a_8*k_{45})+(a_7*a_8*k_{55})
\end{aligned}
$$

$K_{11\ 2}=0.0$
$K_{11\ 3}=(-a_2*k_{22})-(a_4*k_{23})-(a_6*k_{24})-(a_8*k_{25})$
$K_{11\ 4}=0.0$
$K_{11\ 5}=(-a_2*k_{23})-(a_4*k_{33})-(a_6*k_{34})-(a_8*k_{35})$
$K_{11\ 6}=0.0$
$K_{11\ 7}=(-a_2*k_{24})-(a_4*k_{34})-(a_6*k_{44})-(a_8*k_{45})$
$K_{11\ 8}=0.0$
$K_{11\ 9}=(-a_2*k_{25})-(a_4*k_{35})-(a_6*k_{45})-(a_8*k_{55})$
$K_{11\ 10}=0.0$

$$
\begin{aligned}
K_{11\ 11}=&K_{ZZ}^{2B}+(a_2^2*k_{22})+(2*a_4*a_2*k_{23})+(2*a_2*a_6*k_{24})+\\
&(2*a_2*a_8*k_{25})+(a_4^2*k_{33})+(2*a_4*a_6*k_{34})+\\
&(2*a_4*a_8*k_{35})+(a_6^2*k_{44})+(2*a_6*a_8*k_{45})+\\
&(a_8^2*k_{55})+\varepsilon^{2B}z_6^2
\end{aligned}
$$

$K_{11\ 12}=0.0$

$$K_{12\ 1}=0.0$$

$$K_{12\ 2}=(a_1*a_2*k_{22})+(a_2*a_3*k_{23})+(a_1*a_4*k_{23})+(a_2*a_5*k_{24})+$$
$$(a_1*a_6*k_{24})+(a_2*a_7*k_{25})+(a_1*a_8*k_{25})+(a_3*a_4*k_{33})+$$
$$(a_3*a_6*k_{34})+(a_4*a_5*k_{34})+(a_4*a_7*k_{35})+(a_3*a_8*k_{35})+$$
$$(a_5*a_6*k_{44})+(a_6*a_7*k_{45})+(a_5*a_8*k_{45})+(a_7*a_8*k_{55})$$

$$K_{12\ 3}=0.0$$

$$K_{12\ 4}=(-a_2*k_{22})-(a_4*k_{23})-(a_6*k_{24})-(a_8*k_{25})$$

$$K_{12\ 5}=0.0$$

$$K_{12\ 6}=(-a_2*k_{23})-(a_4*k_{33})-(a_6*k_{34})-(a_8*k_{35})$$

$$K_{12\ 7}=0.0$$

$$K_{12\ 8}=(-a_2*k_{24})-(a_4*k_{34})-(a_6*k_{44})-(a_8*k_{45})$$

$$K_{12\ 9}=0.0$$

$$K_{12\ 10}=(-a_2*k_{25})-(a_4*k_{35})-(a_6*k_{45})-(a_8*k_{55})$$

$$K_{12\ 11}=0.0$$

$$K_{12\ 12}=K_{yy}^{2B}+(a_2^2*k_{22})+(2*a_4*a_2*k_{23})+(2*a_2*a_6*k_{24})+$$
$$(2*a_2*a_8*k_{25})+(a_4^2*k_{33})+(2*a_4*a_6*k_{34})+$$
$$(2*a_4*a_8*k_{35})+(a_6^2*k_{44})+(2*a_6*a_8*k_{45})+$$
$$(a_8^2*k_{55})+\varepsilon^{2B}y_6^2$$

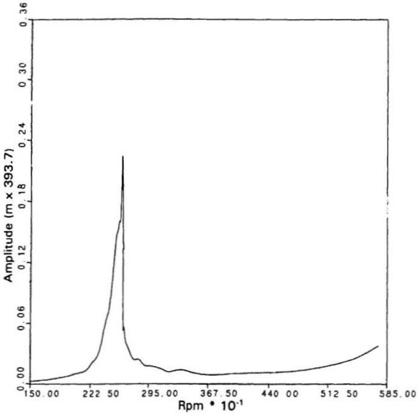
<u>Elements of Force Vector {F}</u>

$$F_1=0$$
$$F_2=0$$
$$F_3=m_2r_2\omega^2\cos(\omega t+\phi_2)$$
$$F_4=m_2r_2\omega^2\sin(\omega t+\phi_2)$$
$$F_5=m_3r_3\omega^2\cos(\omega t+\phi_3)$$
$$F_6=m_3r_3\omega^2\sin(\omega t+\phi_3)$$
$$F_7=m_4r_4\omega^2\cos(\omega t+\phi_4)$$
$$F_8=m_4r_4\omega^2\sin(\omega t+\phi_4)$$

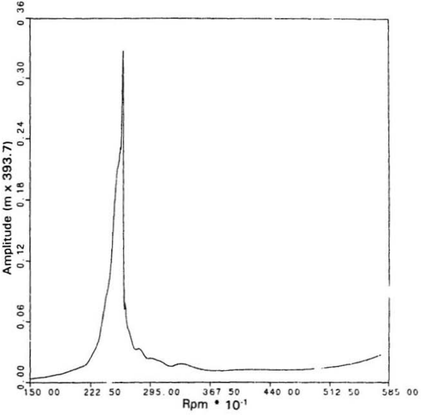
$$F_9=m_5r_5\omega^2\cos(\omega t+\phi_5)$$
$$F_{10}=m_5r_5\omega^2\sin(\omega t+\phi_5)$$
$$F_{11}=0$$
$$F_{12}=0$$

APPENDIX D

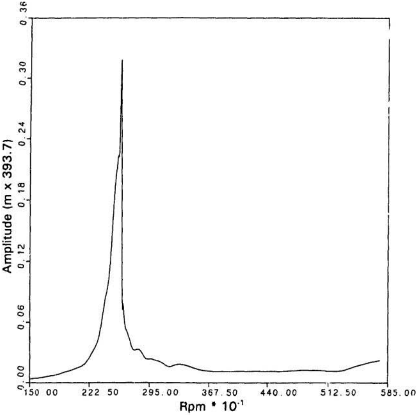
NON-LINEAR UNBALANCED ROTOR RESPONSE DATA



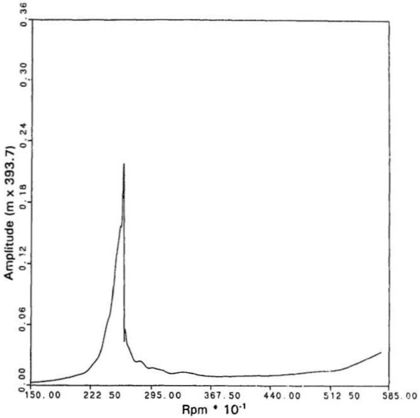
UNBALANCE RESPONSE – ROTOR NO. 1
Mass Station 2 – Vertical Displ.



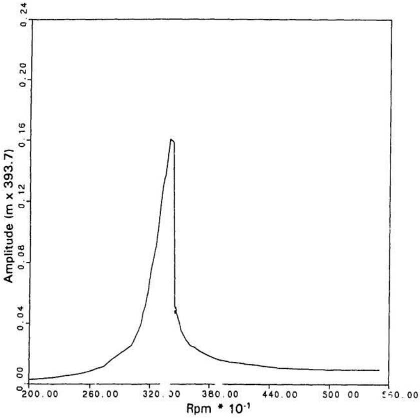
UNBALANCE RESPONSE - ROTOR NO. '
Mass Station 3 - Vertical Displ.



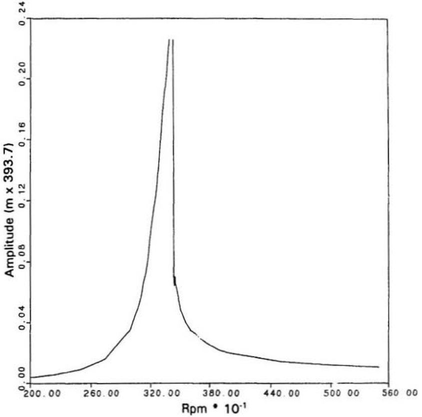
UNBALANCE RESPONSE – ROTOR NO. 1
Mass Station 4 – Vertical Displ.



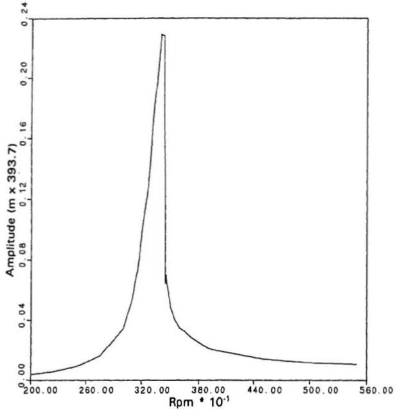
UNBALANCE RESPONSE – ROTOR NO. 1
Mass Station 5 – Vertical Displ.



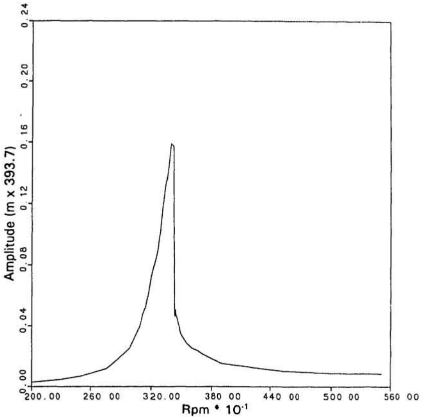
UNBALANCE RESPONSE -- ROTOR NO.2
Mass Station 2 - Vertical Displ.



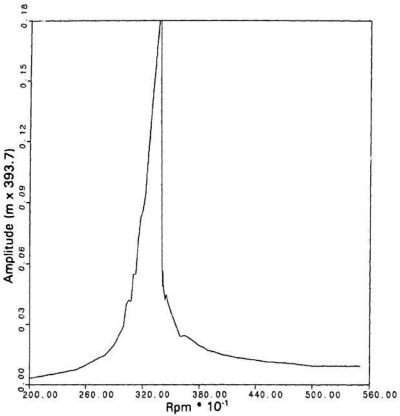
UNBALANCE RESPONSE – ROTOR NO. 2
Mass Station 3 – Vertical Displ.



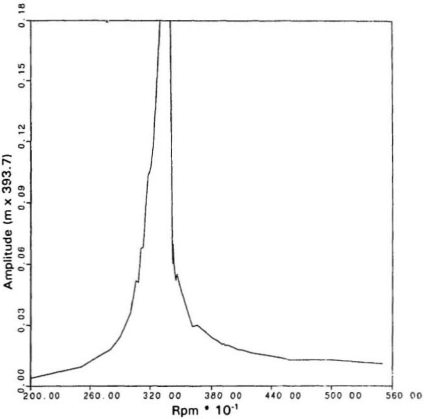
UNBALANCE RESPONSE - ROTOR NO. 2
Mass Station 4 - Vertical Displ.



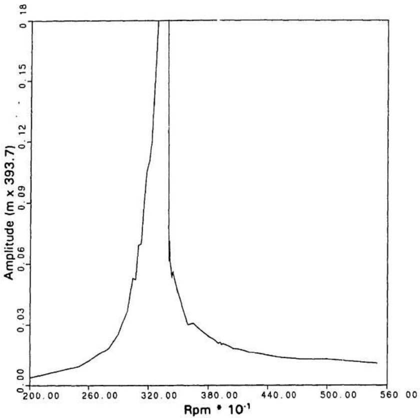
UNBALANCE RESPONSE - ROTOR NO.2
Mass Station 5 - Vertical Displ.



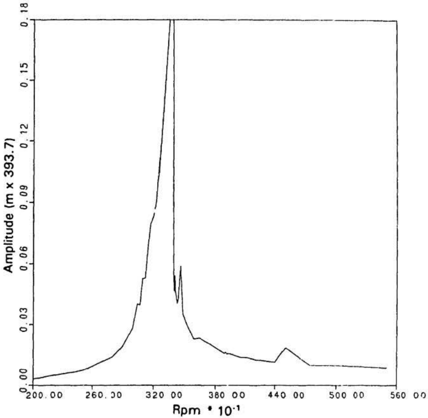
UNBALANCE RESPONSE – ROTOR NO. 3
Mass Station 2 – Vertical Displ.



UNBALANCE RESPONSE - ROTOR NO.3
Mass Station 3 - Vertical Displ.



UNBALANCE RESPONSE – ROTOR NO. 3
Mass Station 4 – Vertical Displ.



UNBALANCE RESPONSE - ROTOR NO. 3
Mass Station 5 - Vertical Displ.

APPENDIX E

FORCE VECTOR DETAILS - IMPEDANCE SOLUTION

In the impedance solution, the harmonic forces are represented as complex numbers. In the case of the rotor unbalance forces, the rotor disk forces in the z and y directions are in the form

$$F_z(t) = Re\left[(F_{o_z}\cos\phi_2 + jF_{o_z}\sin\phi_2)\, e^{j\omega t}\right] \tag{E.1}$$

$$F_y(t) = Re\left[\left(F_{o_y}\cos(\phi_2+\frac{3\pi}{2}) + jF_{o_y}\sin(\phi_2+\frac{3\pi}{2})\right)e^{j\omega t}\right] \tag{E.2}$$

Expanding the terms in Eqs. (E.1) and (E.2) we can show they relate to the harmonic force expressions in the equations of motion. Eq. (E.1) becomes

$$\begin{aligned} F_z(t) &= Re\left[(F_{o_z}\cos\phi_2+jF_{o_z}\sin\phi_2)\,(\cos wt+j\sin wt)\right] \\ &= F_{o_z}\cos\phi_2\cos\omega t-F_{o_z}\sin\phi_2\sin\omega t \\ &= F_{o_z}\cos(\phi_2+\omega t) \end{aligned} \tag{E.3}$$

Expanding Eq. (E.2) it becomes

$$\begin{aligned} F_y(t) &= Re\left[\left(F_{o_y}\cos(\phi_2+\frac{3\pi}{2}) + jF_{o_y}\sin(\phi_2+\frac{3\pi}{2})\right)(\cos\omega t+j\sin\omega t)\right] \\ &= Re\left\{\left\{\left(F_{o_y}\cos\phi_2\cos\frac{3\pi}{2} - F_{o_y}\sin\phi_2\sin\frac{3\pi}{2}\right)+j\left(F_{o_y}\sin\phi_2\cos\frac{3\pi}{2} + \right.\right.\right. \\ &\quad \left.\left.\left. F_{o_y}\cos\phi_2\sin\frac{3\pi}{2}\right)\right\}(\cos\omega t+j\sin\omega t)\right\} \\ &= Re\{(F_{o_y}\sin\phi_2) + j(-F_{o_y}\cos\phi_2)\}(\cos\omega t+j\sin\omega t) \\ &= F_{o_y}\sin\phi_2\cos\omega t+F_{o_y}\cos\phi_2\sin\omega t \\ &= F_{o_y}\sin(\phi_2+\omega t) \end{aligned} \tag{E.4}$$

APPENDIX F

PROGRAM LISTINGS

F.1 Programs For SDOF Non-Linear System

The following Fortran programs were used in the Case Study No.4, of Chapter 2, Forced Vibration of a Single-Degree-of-Freedom-System With Non-Linear Damping. They include a program for the numerical solution of the non-linear problem using the Runge-Kutta method, and programs for equivalent linearization using the optimization method. These programs are typical for the single-degree-of-freedom systems studied in Chapter 2.

F.1.1 NUMERICAL SOLUTION RUNGE-KUTTA METHOD -SDOF SYSTEM

```
C       CASE4.FOR
        REAL Y(2),C(24),W(2,9),X,TOL,XEND,DISP
        EXTERNAL FCN1
        INTEGER N,IND,NW,IER,K,PTS
        OPEN(UNIT=12,FILE='CASE41.DAT',TYPE='NEW')
        N=2
        NW=2
        X=0.0
        Y(1)=0.00387
        Y(2)=0.0
        TOL=0.001
        IND=1
        DO 10 K=1,3000
        XEND=FLOAT(K)*0.001
        CALL DVERK(N,FCN1,X,Y,XEND,TOL,IND,C,NW,W,IER)
        IF(IND.LT.0.OR.IER.GT.0) GO TO 20
        DISP=Y(1)*1000
        WRITE(12,100)X,DISP
   100  FORMAT(6X,F8.3,',',6X,F9.6,',')
    10  CONTINUE
        STOP
    20  CONTINUE
        STOP
        END
C
        SUBROUTINE FCN1(N,X,Y,YPRIME)
        INTEGER N
        REAL Y(N),YPRIME(N),X
        REAL MM,FO,WW,AA,KK,CC,SIGN,DD,BB,FC,CHECK
C
C       SYSTEM VARIABLES
C
        MM=453.23
        FO=4448.2
        WW=37.70
        AA=23007.8
        KK=1789271.
        DD=65404000.
        BB=168635.
        FC=326.09
        CC=FO/MM
C
        IF(Y(2).GT..000001)GO TO 18
        IF(Y(2).LT..000001)GO TO 16
        IF(Y(2).EQ..000001)GO TO 18
    16  SIGN=-1.
        GO TO 19
    18  SIGN=1.
```

```
   19 CONTINUE
C
C      EQUATION 1
C
       YPRIME(1)=Y(2)
C
C      EQUATION OF MOTION 2:VELOCITY SQUARED DAMPING
C
       YPRIME(2)=(CC*SIN(WW*X))-((SIGN)*(AA/MM)
     1 *(Y(2)**2))-((KK/MM)*Y(1))
C
C      EQUATION OF MOTION 1 :VISCOUS DAMPING
C
*      YPRIME(2)=(CC*SIN(WW*X))-((2847.7/MM)*Y(2))
*    1 -((KK/MM)*Y(1))
C
C      EQUATION OF MOTION 3:DISP.SQUARED DAMPING
C
*      YPRIME(2)=(CC*SIN(WW*X))-((SIGN)*(DD/MM)
*    1 *(Y(1)**2))-((KK/MM)*Y(1))
C
C      EQUATION OF MOTION 5:SOLID DAMPING
C
*      YPRIME(2)=(CC*SIN(WW*X))-((SIGN)*(BB/MM)
*    1 *ABS(Y(1)))-((KK/MM)*Y(1))
C
C      EQUATION OF MOTION:COULOMB DAMPING
C
*      CHECK=(FO*SIN(WW*X))-(KK*Y(1))
*      IF((Y(2).EQ.0.0).AND.(CHECK.LT.FC.OR.CHECK.EQ.FC))
*    1 THEN
*      YPRIME(2)=0.0
*      ELSE
*      YPRIME(2)=(CC*SIN(WW*X))-((SIGN)*(FC/MM))
     1 -((KK/MM)*Y(1))
*      END IF
C
C      EQUATION OF MOTION :FREE VIBRATION/COULOMB DAMPING
C
*      YPRIME(2)=((-1)*SIGN*(FC/MM))-((KK/MM)*Y(1))
       RETURN
       END
```

F.1.2 SINGLE VARIABLE OPTIMIZATION

```
C       SOPT4.FOR
        X(1),XSTRT(1),RMAX(1),RMIN(1),PHI(2),PSI(1),W(24)
        N=1
        NCONS=2
        NEQUS=0
        NPENAL=5
        DATA RMAX/ 100000./
        DATA RMIN/ 500./
        DATA XSTRT/ 1000./
        CALL SEEK(N,NCONS,NEQUS,NPENAL,RMAX,RMIN,XSTRT,X,U,
     1  PHI,PSI,NVIOL,W)
        CALL ANSWER(U,X,PHI,PSI,N,NCONS,NEQUS)
        STOP
        END
C
        SUBROUTINE UREAL(X,U)
        DIMENSION T(200),X(1),XR(200),XL(200),UD(200)
        REAL MA,WSQ,FO,XO,THETA,KK
        OPEN(UNIT=12,FILE='EXAMPLE.DAT',TYPE='OLD')
        REWIND 12
        NN=167
        DO 60 II=1,NN
        READ(12,*)T(II),XR(II)
   50   FORMAT(F8.3,F9.6)
   60   CONTINUE
           CLOSE(12)
        DO 65 II=1,NN
        XR(II)=XR(II)*.001
   65   CONTINUE
        MA=453.23
        WSQ=(37.70)**2
        FO=4448.2
        KK=1789271.
        XO=FO/((((KK-(MA*WSQ))**2)+((X(1)**2)*WSQ))**0.5)
        THETA=ATAN((X(1)*(WSQ**0.5))/(KK-(MA*WSQ)))
        DO 100 II=1,NN
        XL(II)=XO*SIN(((WSQ**0.5)*T(II))-THETA)
  100   CONTINUE
        DO 200 JJ=1,NN
        UD(JJ)=(XR(JJ)-XL(JJ))**2
  200   CONTINUE
        U=UD(1)
        DO 205 II=1,NN
        JJ=II+1
        U=U+UD(JJ)
  205   CONTINUE
        RETURN
        END
C
```

```
      SUBROUTINE CONST(X,NCONS,PHI)
      DIMENSION X(1),PHI(1)
      PHI(1)=100000.-X(1)
      PHI(2)=X(1)-500.
      RETURN
      END
C
      SUBROUTINE EQUAL(X,PSI,NEQUS)
      DIMENSION X(1),PSI(1)
      RETURN
      END
```

F.1.3 DUAL VARIABLE OPTIMIZATIOM

```
C       DOPT4.FOR
        DIMENSION X(2),XSTRT(2),RMAX(2),RMIN(2),PHI(4),
      1 PSI(1),W(24)
        N=2
        NCONS=4
        NEQUS=0
        NPENAL=5
        DATA RMAX/ 500000.,3000000./
        DATA RMIN/ 500.,100000./
        DATA XSTRT/ 1000.,1500000./
        CALL SEEK(N,NCONS,NEQUS,NPENAL,RMAX,RMIN,XSTRT,X,U,
      1 PHI,PSI,NVIOL,W)
        CALL ANSWER(U,X,PHI,PSI,N,NCONS,NEQUS)
        STOP
        END
C
        SUBROUTINE UREAL(X,U)
        DIMENSION T(200),X(1),XR(200),XL(200),UD(200),
      1 CHECK(200)
        REAL MA,WSQ,FO,XO,THETA,FACTOR
        OPEN(UNIT=10,FILE='EXAMPLE.DAT',TYPE='OLD')
        REWIND 10
        NN=167
        DO 60 II=1,NN
        READ(10,*)T(II),XR(II)
   60   CONTINUE
        CLOSE(10)
        DO 65 II=1,NN
        XR(II)=XR(II)*.001
   65   CONTINUE
        MA=453.23
        WSQ=(37.70)**2
        FO=4448.2
        XO=FO/((((X(2)-(MA*WSQ))**2)+((X(1)**2)*WSQ))**0.5)
        THETA=ATAN((X(1)*(WSQ**0.5))/(X(2)-(MA*WSQ)))
        DO 100 II=1,NN
        XL(II)=XO*SIN(((WSQ**0.5)*T(II))-THETA)
  100   CONTINUE
        DO 200 JJ=1,NN
        UD(JJ)=(XR(JJ)-XL(JJ))**2
  200   CONTINUE
        U=UD(1)
        DO 205 II=1,NN
        JJ=II+1
        U=U+UD(JJ)
  205   CONTINUE
        RETURN
        END
C
```

```
      SUBROUTINE CONST(X,NCONS,PHI)
      DIMENSION X(1),PHI(1)
      PHI(1)=500000.-X(1)
      PHI(2)=X(1)-500.
      PHI(3)=3000000.-X(2)
      PHI(4)=X(2)-100000.
      RETURN
      END
C
      SUBROUTINE EQUAL(X,PSI,NEQUS)
      DIMENSION X(1),PSI(1)
      RETURN
      END
```

F.2 MULTI-DEGREE-OF-FREEDOM SYSTEM WITH NON-LINEAR DAMPING

The following Fortran programs were used in the Case Study No.5, Free Vibration of a Multi-Degree-of-Freedom System With Non-Linear Damping. They include a program for the numerical solution of the non-linear system of equations using the Runge-Kutta method, and a program for the equivalent linearization using the optimization method. In the linearization program, modal analysis is used to solve the system response in the subroutine Ureal.

F.2.1 NUMERICAL SOLUTION - MDOF DAMPING

```
C       CASE 5.FOR
C       NUMERICAL SOLUTION - RUNGE-KUTTA
        REAL Y(6),C(24),W(6,9),X,TOL,XEND
        EXTERNAL FCN1
        INTEGER N,IND,NW,IER,K
        OPEN(UNIT=16,FILE='OUTPUT.DAT',TYPE='NEW')
        OPEN(UNIT=15,FILE='X1.DAT',TYPE='NEW')
        OPEN(UNIT=14,FILE='X2.DAT',TYPE='NEW')
        OPEN(UNIT=13,FILE='X3.DAT',TYPE='NEW')
        N=6
        NW=6
C
C       INITIAL CONDITIONS
C
        Y(1)=0.040
        Y(2)=0.0
        Y(3)=0.030
        Y(4)=0.0
        Y(5)=0.020
        Y(6)=0.0
C
        TOL=0.001
        IND=1
        DO 10 K=1,1500
        XEND=FLOAT(K)*0.01
        CALL DVERK(N,FCN1,X,Y,XEND,TOL,IND,C,NW,W,IER)
        IF(IND.LT.0.OR.IER.GT.0) GO TO 20
        WRITE(16,100)X,Y(1),Y(3),Y(5)
 100    FORMAT(F8.4,',',1X,F8.4,',',F8.4,',',F8.4,',')
        WRITE(15,110)X,Y(1)
 110    FORMAT(F8.4,',',1X,F8.4,',')
        WRITE(14,110)X,Y(3)
        WRITE(13,110)X,Y(5)
  10    CONTINUE
        STOP
  20    CONTINUE
        STOP
        END
C
        SUBROUTINE FCN1(N,X,Y,YPRIME)
        INTEGER N
        REAL Y(N),YPRIME(N),X
        REAL K1,K2,K3,K4,M1,M2,M3,C1,C2,C3,C4,C5,C6
C
C       INPUT SYSTEM VARIABLES
C       SYSTEM HAS THE FOLLOWING NON LINEAR DAMPING FORCES
C
C       C1,C2,C3 - VELOCITY SQUARED DAMPING
C                  DAMPING FORCE=(SIGN DX/DT)*C*(DX/DT)**2
```

```
C       C4,C5       - DISPLACEMENT SQUARED DAMPING
C                     DAMPING FORCE=(SIGN DX/DT)*C*(X**2)
C       C6          - COULOMB DAMPING
C                     DAMPING FORCE=(SIGN DX/DT)*(NORMAL FORCE)
        K1=1789000.
        K2=1400000.
        K3=1000000.
        K4=2000000.
        M1=450.
        M2=430.
        M3=100.
        C1=45000.
        C2=27000.
        C3=55000.
        C4=9000000.
        C5=6000000.
        C6=0.0
        IF(Y(2).GT..000001)GO TO 30
        IF(Y(2).LT..000001)GO TO 25
        IF(Y(2).EQ..000001)GO TO 30
   25   SIGN1=-1.0
        GO TO 35
   30   SIGN1=1.0
   35   CONTINUE
        IF(Y(4).GT..000001)GO TO 50
        IF(Y(4).LT..000001)GO TO 40
        IF(Y(4).EQ..000001)GO TO 50
   40   SIGN2=-1.0
        GO TO 55
   50   SIGN2=1.0
   55   CONTINUE
        IF(Y(6).GT..000001)GO TO 70
        IF(Y(6).LT..000001)GO TO 60
        IF(Y(6).EQ..000001)GO TO 70
   60   SIGN3=-1.0
        GO TO 75
   70   SIGN3=1.0
   75   CONTINUE
        YPRIME(1)=Y(2)
        YPRIME(2)=((((-1)*(K1+K2))/M1)*Y(1))+((K2/M1)
      1 *Y(3))-((C1/M1)*SIGN1*(Y(2)**2))-((C2/M1)*SIGN1
      1 *(Y(2)**2))+((C2/M1)*SIGN2* (Y(4)**2))-((C5/M1)*SIGN1
      1 *(Y(1)**2))+((C5/M1)*SIGN3*(Y(5)**2))
        YPRIME(3)=Y(4)
        YPRIME(4)=((K2/M2)*Y(1))-(((K2+K3)/M2)*Y(3))
      1 +((K3/M2)*Y(5))+((C2/M2)*SIGN1*(Y(2)**2))
      1 -((C2/M2)*SIGN2*(Y(4)**2))-((C3/M2 )*SIGN2
      1 *(Y(4)**2))-((C6/M2)*SIGN2)+((C3/M2)*SIGN3*(Y(6)**2)
        YPRIME(5)=Y(6)
        YPRIME(6)=((K3/M3)*Y(3))-(((K3+K4)/M3)*Y(5)+((C5
      1 /M3)*SIGN1*(Y(1)**2))+((C3/M3)*SIGN2*(Y(4)**2))-((C
      1 3/M3)*SIGN3*(Y(6)**2))-((C4/M3)*SIGN3*(Y(5)**2))
```

```
      1 -((C5/M3)*SIGN3*(Y(5)**2))
        RETURN
        END
```

F.2.2 OPTIMIZATION PROGRAM - MDOF

```
C       OPT5.FOR
C       FREE VIBRATION OF A MULTI-DEGREE-OF-FREEDOM SYSTEM
C       WITH NON-LINEAR DAMPING
        DIMENSIONX(5),XSTRT(5),RMAX(5),RMIN(5),PHI(12),
      1 PSI(1),W(1000)
        N=5
        NCONS=10
        NEQUS=0
        NPENAL=5
        DATA RMAX/5000000.0,5000000.0,5000000.0,5000000.0,
      1 5000000.0/
        DATA RMIN/1.,1.,1.,1.,1./
        DATA XSTRT/5000.,3000.,6500.,4500.,2649./
        CALL DAVID(N,NCONS,NEQUS,NPENAL,RMAX,RMIN,
      1 XSTRT,X,U,PHI,PSI,NVIOL,W)
        CALL ANSWER(U,X,PHI,PSI,N,NCONS,NEQUS)
        STOP
        END
C
        SUBROUTINE UREAL(X,U)
        DIMENSION X(1)
        DIMENSIONXM(3,3),XSTIF(3,3),XD(3,3),D(6,6),XSINV(3,3)
        DIMENSION D3(3,3),D4(3,3),A(3,3),C(3,3),R(3,3),WK(1000)
        DIMENSION WGK(1000),EVR(6),EVI(6),ZVR(6,6),ZVI(6,6)
        DIMENSION RZIN(6,6),CZIN(6,6),XORG(6,1),ZOR(6),ZOI(6)
        DIMENSION T(350),CKR(6,6),CKI(6,6)
        DIMENSION EX(6),XXR(200,6),XXI(200,6)
        DIMENSION TIME(200),UD(200),XNUM1(200),XNUM2(200),
      1 XNUM3(200)
        REAL M1,M2,M3,C1,C2,C3,C4,C5,C6,K1,K2,K3,K4,TT,INCR
        COMPLEX H(6,6),EV(6),ZV(6,6),ZINV(6,6),ZO(6,1)
        COMPLEX Z(200,6),TRIG(200,6),S
        COMPLEX EXX(6),ZZ(6,1),XXX(6,1),XX(200,6),XOO(200,6)
        COMPLEX CK(6,6)
C
C       INPUT NUMERICAL SOLUTION
C
        OPEN(UNIT=10,FILE='EXAMPLE.DAT',TYPE='OLD')
        REWIND 10
        DATA=40
        DO 5 II=1,DATA
        READ(10,*)TIME(II),XNUM1(II),XNUM2(II),XNUM3(II)
 5      CONTINUE
        TSTART=0.90
```

```
      N=3
C
C     INPUT THE INITIAL CONDITIONS
C
      XORG(1,1)=0.0
      XORG(2,1)=0.0
      XORG(3,1)=0.0
      XORG(4,1)=0.020
      XORG(5,1)=0.015
      XORG(6,1)=0.015
C
C     INPUT DATA FOR XM-MASS MATRIX,XS-STIFFNESS MATRIX
C     AND XD-DAMPING MATRIX
C
      M1=4500.0
      M2=4300.0
      M3=1000.0
      C1=X(1)
      C2=X(2)
      C3=X(3)
      C4=X(4)
      C5=X(5)
      C6=0.0
      K1=1789000.
      K2=1400000.
      K3=1000000.
      K4=2000000.
      XM(1,1)=M1
      XM(1,2)=0.0
      XM(1,3)=0.0
      XM(2,1)=0.0
      XM(2,2)=M2
      XM(3,1)=0.0
      XM(3,2)=0.0
      XM(3,3)=M3
      XSTIF(1,1)=K1+K2
      XSTIF(1,2)=-K2
      XSTIF(1,3)=0.0
      XSTIF(2,1)=-K2
      XSTIF(2,2)=K2+K3
      XSTIF(2,3)=-K3
      XSTIF(3,1)=0.0
      XSTIF(3,2)=-K3
      XSTIF(3,3)=K3+K4
      XD(1,1)=C1+C2+C5
      XD(1,2)=-C2
      XD(1,3)=-C5
      XD(2,1)=-C2
      XD(2,2)=C2+C3+C6
      XD(2,3)=-C3
      XD(3,1)=-C5
      XD(3,2)=-C3
```

```
      XD(3,3)=C3+C4+C5
C
      IA=3
      IDGT=3
      CALL LINV2F(XSTIF,N,IA,XSINV,IDGT,WK,IER)
      DO 20 I=1,N
      DO 20 J=1,N
 20   XSINV(I,J)=-1.0*XSINV(I,J)
      M=3
      L=3
      CALL RGMPRD(XSINV,XM,D3,N,M,L)
      CALL RGMPRD(XSINV,XD,D4,N,M,L)
C
C     DYNAMIC MATRIX IS CALCULATED
C
      NN=2*N
      DO 200 I=1,NN
      DO 200 J=1,NN
200   D(I,J)=0.0
      MMM=N
      DO 210 I=1,N
      DO 210 J=1,N
      JJ=J+MMM
      II=I+MMM
      D(II,J)=0.0
      D(II,J)=D3(I,J)
 210  D(II,JJ)=D4(I,J)
      DO 220 I=1,N
      II=I+N
 220  D(I,II)=1.0
C
C   CALCULATE THE EIGENVALUES AND EIGENVECTORS
C
      DO 240 I=1,NN
      DO 240 J=1,NN
240   H(I,J)=CMPLX(D(I,J),0.0)
      CALL EIGCC(H,NN,NN,2,EV,ZV,NN,WGK,IER)
      S=CMPLX(1.0,0.0)
      DO 250 I=1,NN
 250  EV(I)=S/EV(I)
      DO 255 I=1,NN
      EVR(I)=REAL(EV(I))
      EVI(I)=AIMAG(EV(I))
 255  CONTINUE
C
      CALL CINV(ZV,ZINV,6)
C
C     OBTAIN   THE   INITIAL   CONDITIONS   IN   THE   NEW   STATE
C     VARIABLE
C
      DO 366 I=1,NN
      XOO(I,1)=CMPLX(XORG(I,1),0.0)
```

```
 366    CONTINUE
        CALL COMPM(ZINV,XOO,ZO,6,6,1)
C
C       FIND THE RESPONSE OF THE SYSTEM
C
        INCR=0.01
        TT=TSTART
        DO 550 II=1,DATA
        DO 555 I=1,NN
        T(II)=TT
        TRIG(II,I)=CMPLX(COS(EVI(I)*T(II)),SIN(EVI(I)*T(II)))
 555    CONTINUE
        TT=TT+INCR
 550    CONTINUE
        DO 610 II=1,DATA
        DO 580 I=1,NN
        EX(I)=EXP(EVR(I)*T(II))
        EXX(I)=CMPLX(EX(I),0.0)
        Z(II,I)=TRIG(II,I)*EXX(I)*ZO(I,1)
        ZZ(I,1)=Z(II,I)
 580    CONTINUE
        CALL COMPM(ZV,ZZ,XXX,6,6,1)
        DO 600 I=1,NN
 600    XX(II,I)=XXX(I,1)
 610    CONTINUE
        DO 650 II=1,DATA
        DO 650 I=1,NN
        XXR(II,I)=REAL(XX(II,I))
        XXI(II,I)=AIMAG(XX(II,I))
 650    CONTINUE
C
C       CALCULATE THE VALUE OF THE OPTIMIZATION FUNCTION U
C
        DO 700 II=1,DATA
        UD(II)=((XNUM1(II)-XXR(II,4))**2)+((XNUM2(II)
      1 -XXR(II,5))**2)+((XNUM3(II)-XXR(II,6))**2)
 700    CONTINUE
        UU=UD(1)
        DO 705 II=1,DATA
        JJ=II+1
        UU=UU+UD(JJ)
 705    CONTINUE
        U=1.0E12*UU
        RETURN
        END
C
        SUBROUTINE CONST(X,NCONS,PHI)
        DIMENSION X(1),PHI(1)
        PHI(1)=5000000.-X(1)
        PHI(2)=X(1)-1.0
        PHI(3)=5000000.-X(2)
        PHI(4)=X(2)-1.0
```

```
      PHI(5)=5000000.-X(3)
      PHI(6)=X(3)-1.0
      PHI(7)=5000000.-X(4)
      PHI(8)=X(4)-1.
      PHI(9)=5000000.-X(5)
      PHI(10)=X(5)-1.
      RETURN
      END
C
      SUBROUTINE EQUAL(X,PSI,NEQUS)
      DIMENSION X(1),PSI(1)
      RETURN
      END
C
      SUBROUTINE RGMPRD(A,C,R,N,M,L)
      REAL A(1),C(1),R(1)
      IR=0
      IK=-M
      DO 13 K=1,L
      IK=IK+M
      DO 13 J=1,N
      IR=IR+1
      JI=J-N
      IC=IK
      R(IR)=0
      DO 13 I=1,M
      JI=JI+N
      IC=IC+1
13    R(IR)=R(IR)+A(JI)*C(IC)
      RETURN
      END
C
      SUBROUTINE COMPM(A,C,R,N,M,L)
      COMPLEX A(1),C(1),R(1)
      IR=0
      IK=-M
      DO 13 K=1,L
      IK=IK+M
      DO 13 J=1,N
      IR=IR+1
      JI=J-N
      IC=IK
      R(IR)=0
      DO 13 I=1,M
      JI=JI+N
      IC=IC+1
 13   R(IR)=R(IR)+A(JI)*C(IC)
      RETURN
      END
C
      SUBROUTINE CINV(H,HINVS,N)
      COMPLEX H(6,6),HINVS(6,6),A(6,6),B(6,6),SUM
```

```
      CALL CSUBN(H,A,N)
      NM1=N-1
      DO 40 I=1,NM1
      SUM=0
      DO 41 K=1,N
 41   SUM =SUM+A(K,K)
      SUM=SUM/I
      DO 42 J=1,N
 42   A(J,J)=A(J,J)-SUM
      IF(I.EQ.NM1) CALL CSUBN(A,HINVS,N)
      CALL CMPLY(H,A,B,N)
 40   CALL CSUBN(B,A,N)
      DO 43 I=1,N
      DO 43 J=1,N
 43   HINVS(I,J)=HINVS(I,J)/A(1,1)
      RETURN
      END
C
      SUBROUTINE CMPLY(A,B,C,N)
      COMPLEX A(6,6),B(6,6),C(6,6)
      DO 40 I=1,N
      DO 40 J=1,N
      C(I,J)=0
      DO 40 K=1,N
 40   C(I,J)=C(I,J)+A(I,K)*B(K,J)
      RETURN
      END
C
      SUBROUTINE CSUBN(A,B,N)
      COMPLEX A(6,6),B(6,6)
      DO 40 I=1,N
      DO 40 J=1,N
 40   B(I,J)=A(I,J)
      RETURN
      END
```

F.3 FLEXIBLE ROTOR SUPPORTED ON BEARINGS WITH NON-LINEAR FLEXIBILITY

The following Fortran programs were used in the balancing of the rotors studied in Chapter 3. The first program, NUM6.FOR, calculates the response of the non-linear rotor system equations of motion, using the Newmark-β Method in conjunction with an iteration procedure. The program OPT12.FOR finds the equivalent linear bearing stiffness coefficients using the optimization principles, while the program BAL7.FOR calculates the balance weights for the equivalent linear rotor system.

F.3.1 NUMERICAL SOLUTION - NON-LINEAR ROTOR SYSTEM

```
C       NUM6.FOR
C       CALCULATES THE RESPONSE OF A MDOF NON-LINEAR ROTOR
C          SYSTEM USING AN ITERATION METHOD IN CONJUNCTION
C       WITH THE NEWMARK METHOD
        REAL M3,M4,PI
        REAL THETA2,THETA3,THETA4,THETA5,EC2,EC3,EC4,EC5
        REAL RPM,W,DW
        REAL M(12,12),C(12,12),K(12,12),MINV(12,12)
        REAL ZI(12),ZDI(12),ZDDI(12)
        REAL T,F(12),FI(12)
        REAL Z(12),ZD(12),ZDD(12),AMP(12)
        REAL MM(12,12),CC(12,12),DD(12,12),GG(12,12),S(12,12)
        REAL V1(12),V2(12),W1(12),W2(12),W3(12),W4(12),W5(12)
        REAL ALPHA,BETA,DELT,TSTART,CONST1,CONST2
        REAL ZA(12),ZAI(12),DIFF(12)
        REAL EB1,EB2,TOL,KLIN1,KLIN2,KLIN3,KLIN4
        REAL MB11,MB21,MB31,MB12,MB22,MB32
        REAL ECC11,ECC21,ECC31,ECC12,ECC22,ECC32
        REAL BETA11,BETA21,BETA31,BETA12,BETA22,BETA32
        COMPLEX GGC(12,12),GGCI(12,12)
        OPEN(UNIT=16,FILE='STIFF.DAT',TYPE='OLD')
        OPEN(UNIT=14,FILE='DAMP.DAT',TYPE='OLD')
        OPEN(UNIT=12,FILE='MASS.DAT',TYPE='OLD')
        OPEN(UNIT=10,FILE='DAT.DAT',TYPE='NEW')
C
C       INPUT SYSTEM VARIABLES
C
        ALPHA=1.0/4.0
        BETA=0.5
        DELT=0.0001
        TOL=0.05
        EB1=1.0e14
        EB2=1.0e14
        N=12
        DO 3 I=1,N
        DO 3 J=1,N
        READ(16,*)K(I,J)
        READ(14,*)C(I,J)
        READ(12,*)M(I,J)
    3   CONTINUE
        REWIND 16
        REWIND 14
        REWIND 12
        PI=3.14159
        THETA2=0.785
        THETA3=0.785
        THETA4=2.355
        THETA5=2.355
```

```
      EC2=0.00002
      EC3=0.00002
      EC4=0.00002
      EC5=0.00002
      M2=30.0
      M3=30.0
      M4=30.0
      M5=30.0
      DO 39 I=1,12
   39 WRITE(10,*)M(I,1),M(I,2),M(I,3),M(I,4),M(I,5),M(I,6)
    1 ,M(I,7),M(I,8),M(I,9),M(I,10),M(I,11),M(I,12)
      DO 38 I=1,12
   38 WRITE(10,*)K(I,1),K(I,2),K(I,3),K(I,4),K(I,5),K(I,6)
     1 ,K(I,7),K(I,8),K(I,9),K(I,10),K(I,11),K(I,12)
      KLIN1=K(1,1)
      KLIN2=K(2,2)
      KLIN3=K(11,11)
      KLIN4=K(12,12)
      DO 37 I=1,12
   37 WRITE(10,*)C(I,1),C(I,2),C(I,3),C(I,4),C(I,5),C(I,6)
     1 ,C(I,7),C(I,8),C(I,9),C(I,10),C(I,11),C(I,12)
      RPM=6160
      W=(RPM*(2.0*PI))/60.0
      WRITE(10,*)W
C
C     INPUT INITIALS CONDITIONS
C
      TSTART=0.0
      T=TSTART
      DO 5 I=1,N
      ZI(I)=0.0
      ZDI(I)=0.0
    5 CONTINUE
      DO 10 I=1,N
      Z(I)=ZI(I)
      ZD(I)=ZDI(I)
   10 CONTINUE
C
      CALL LINDS(N,M,N,MINV,N)
C
C     INITIAL UNBALANCE FORCES AND
C     BALANCE WEIGHT FORCE COMPONENTS
C
      MB11=0.01800*0.0
      MB21=0.02180*0.0
      MB31=0.25285*0.0
      MB12=0.0
      MB22=0.0
      MB32=0.0
      ECC11=0.152*0.0
      ECC21=0.152*0.0
      ECC31=0.152*0.0
```

```
      ECC12=0.0
      ECC22=0.0
      ECC32=0.0
      BETA11=-2.851659*0.0
      BETA21=-1.593565*0.0
      BETA31=-0.788303*0.0
      BETA12=0.0
      BETA22=0.0
      BETA32=0.0
      FI(1)=0.0
      FI(2)=0.0
      FI(3)=0.0+(MB11*ECC11*(W**2)*COS((W*T)+BETA11))+
     1 (M2*EC2*(W**2)*COS((W*T)+THETA2))
     1 +(MB12*ECC12*(W**2)*COS((W*T)+BETA12))
      FI(4)=0.0+(MB11*ECC11*(W**2)*sin((W*T)+BETA11))+
     1 (M2*EC2*(W**2)*SIN((W*T)+THETA2))
     1 +(MB12*ECC12*(W**2)*SIN((W*T)+BETA12))
      FI(5)=M3*EC3*(W**2)*COS((W*T)+THETA3)
      FI(6)=M3*EC3*(W**2)*sin((W*T)+THETA3)
      FI(7)=M4*EC4*(W**2)*COS((W*T)+THETA4)
     1 +(MB21*ECC21*(W**2)*COS((W*T)+BETA21))
     1 +(M322*ECC22*(W**2)*COS((W*T)+BETA22))
      FI(8)=M4*EC4*(W**2)*sin((W*T)+THEta4)
     1 +(MB21*ECC21*(W**2)*SIN((W*T)+BETA21))
     1 +(MB22*ECC22*(W**2)*SIN((W*T)+BETA22))
      FI(9)=0.0+(MB31*ECC31*(W**2)*COS((W*T)+BETA31))+
     1 (M5*EC5*(W**2)*COS((W*T)+THETA5))
     1 +(MB32*ECC32*(W**2)*COS((W*T)+BETA32))
      FI(10)=0.0+(MB31*ECC31*(W**2)*sin((W*T)+BETA31))+
     1 (M5*EC5*(W**2)*SIN((W*T)+THETA5))
     1 +(MB32*ECC32*(W**2)*SIN((W*T)+BETA32))
      FI(11)=0.0
      FI(12)=0.0
      CALL MRRRR(N,N,MINV,N,N,1,FI,N,N,1,ZDDI,N)
      DO 11 I=1,N
      ZDD(I)=ZDDI(I)
   11 CONTINUE
C
C     INPUT STARTING VALUES FOR ASSUMED DISPLACEMENTS
C
      ZAI(1)=2.0E-07
      ZAI(2)=-1.0E-06
      ZAI(3)=1.5E-06
      ZAI(4)=1.0E-06
      ZAI(5)=7.0E-07
      ZAI(6)=2.0E-06
      ZAI(7)=-7.0E-08
      ZAI(8)=2.0E-07
      ZAI(9)=1.0E-06
      ZAI(10)=1.5E-06
      ZAI(11)=-1.0E-06
      ZAI(12)=2.0E-07
```

```
C
      ITER=20000
      STEP=0
      DO 100 I=1,ITER
      STEP=STEP+1
      T=T+DELT


C
C     CALCULATE FORCE VECTOR AT T=T+DELT
C
      F(1)=0.0
      F(2)=0.0
      F(3)=0.0+(MB11*ECC11*(W**2)*COS((W*T)+BETA11))+
     1 (M2*EC2*(W**2)*COS((W*T)+THETA2))
     1 +(MB12*ECC12*(W**2)*COS((W*T)+BETA12))
      F(4)=0.0+(MB11*ECC11*(W**2)*sin((W*T)+BETA11))+
     1 (M2*EC2*(W**2)*SIN((W*T)+THETA2))
     1 +(MB12*ECC12*(W**2)*SIN((W*T)+BETA12))
      F(5)=M3*EC3*(W**2)*COS((W*T)+THETA3)
      F(6)=M3*EC3*(W**2)*sin((W*T)+THETA3)
      F(7)=M4*EC4*(W**2)*COS((W*T)+THETA4)
     1 +(MB21*ECC21*(W**2)*COS((W*T)+BETA21))
     1 +(MB22*ECC22*(W**2)*COS((W*T)+BETA22))
      F(8)=M4*EC4*(W**2)*sin((W*T)+THEta4)
     1 +(MB21*ECC21*(W**2)*SIN((W*T)+BETA21))
     1 +(MB22*ECC22*(W**2)*SIN((W*T)+BETA22))
      F(9)=0.0+(MB31*ECC31*(W**2)*COS((W*T)+BETA31))+
     1 (M5*EC5*(W**2)*COS((W*T)+THETA5))
     1 +(MB32*ECC32*(W**2)*COS((W*T)+BETA32))
      F(10)=0.0+(MB31*ECC31*(W**2)*sin((W*T)+BETA31))+
     1 (M5*EC5*(W**2)*SIN((W*T)+THETA5))
     1 +(MB32*ECC32*(W**2)*SIN((W*T)+BETA32))
      F(11)=0.0
      F(12)=0.0
  200 CONTINUE
C
C     ASSUME DISPLACEMENTS
C
      ZA(1)=ZAI(1)
      ZA(2)=ZAI(2)
      ZA(3)=ZAI(3)
      ZA(4)=ZAI(4)
      ZA(5)=ZAI(5)
      ZA(6)=ZAI(6)
      ZA(7)=ZAI(7)
      ZA(8)=ZAI(8)
      ZA(9)=ZAI(9)
      ZA(10)=ZAI(10)
      ZA(11)=ZAI(11)
      ZA(12)=ZAI(12)
C
```

```
C      CALCULATE THE NON-LINEAR TERMS OF [K]
C
       K(1,1)=KLIN1+(EB1*(ABS(ZA(1))**2))
       K(2,2)=KLIN2+(EB1*(ABS(ZA(2))**2))
       K(11,11)=KLIN3+(EB2*(ABS(ZA(11))**2))
       K(12,12)=KLIN4+(EB2*(ABS(ZA(12))**2))
C
C      CALCULATE RESPONSE
C
       CONST1=1.0/(ALPHA*(DELT**2))
       CONST2=BETA/(ALPHA*DELT)
       DO 40 KK=1,N
       DO 40 J=1,N
       MM(KK,J)=CONST1*M(KK,J)
       CC(KK,J)=CONST2*C(KK,J)
   40  CONTINUE
       DO 41 KK=1,N
       DO 41 J=1,N
       DD(KK,J)=MM(KK,J)+CC(KK,J)
   41  CONTINUE
       DO 42 KK=1,N
       DO 42 J=1,N
       GG(KK,J)=DD(KK,J)+K(KK,J)
   42  CONTINUE
       DO 44 KK=1,N
       DO 44 J=1,N
       GGC(KK,J)=CMPLX(GG(KK,J),0.0)
  44   CONTINUE
       CALL LINCG(N,GGC,N,GGCI,N)
       DO 45 KK=1,N
       DO 45 J=1,N
       S(KK,J)=REAL(GGCI(KK,J))
   45  CONTINUE
       DO 50 KK=1,N
       V1(KK)=((1.0/(ALPHA*(DELT**2)))*ZI(KK))+((1.0/
     1 (ALPHA*DELT))*ZDI(KK))+(((1.0/(2.0*ALPHA))-1.0)*
     1 ZDDI(KK))
       V2(KK)=((BETA/(ALPHA*DELT))*ZI(KK))+(((BETA/ALPHA)
     1 -1.0)*ZDI(KK))+(((BETA/ALPHA)-2.0)*(DELT/2.0)
     1 *ZDDI(KK))
   50  CONTINUE
       CALL MRRRR(N,N,M,N,N,1,V1,N,N,1,W1,N)
       CALL MRRRR(N,N,C,N,N,1,V2,N,N,1,W2,N)
       DO 60 KK=1,N
       W3(KK)=F(KK)+W1(KK)
       W4(KK)=W3(KK)+W2(KK)
   60  CONTINUE
       CALL MRRRR(N,N,S,N,N,1,W4,N,N,1,W5,N)
       DO 70 KK=1,N
       Z(KK)=W5(KK)
       AMP(KK)=Z(KK)*(1000.0/25.4)
   70  CONTINUE
```

```
C
C     CHECK ASSUMED VALUES VS. THE CALCULATED VALUE
C
      DIFF(1)=(Z(1)-ZA(1))/Z(1)
      IF(ABS(DIFF(1)).GT.TOL)GO TO 250
      DIFF(2)=(Z(2)-ZA(2))/Z(2)
      IF(ABS(DIFF(2)).GT.TOL)GO TO 250
      DIFF(3)=(Z(3)-ZA(3))/Z(3)
      IF(ABS(DIFF(3)).GT.TOL)GO TO 250
      DIFF(4)=(Z(4)-ZA(4))/Z(4)
      IF(ABS(DIFF(4)).GT.TOL)GO TO 250
      DIFF(5)=(Z(5)-ZA(5))/Z(5)
      IF(ABS(DIFF(5)).GT.TOL)GO TO 250
      DIFF(6)=(Z(6)-ZA(6))/Z(6)
      IF(ABS(DIFF(6)).GT.TOL)GO TO 250
      DIFF(7)=(Z(7)-ZA(7))/Z(7)
      IF(ABS(DIFF(7)).GT.TOL)GO TO 250
      DIFF(8)=(Z(8)-ZA(8))/Z(8)
      IF(ABS(DIFF(8)).GT.TOL)GO TO 250
      DIFF(9)=(Z(9)-ZA(9))/Z(9)
      IF(ABS(DIFF(9)).GT.TOL)GO TO 250
      DIFF(10)=(Z(10)-ZA(10))/Z(10)
      IF(ABS(DIFF(10)).GT.TOL)GO TO 250
      DIFF(11)=(Z(11)-ZA(11))/Z(11)
      IF(ABS(DIFF(11)).GT.TOL)GO TO 250
      DIFF(12)=(Z(12)-ZA(12))/Z(12)
      IF(ABS(DIFF(12)).GT.TOL)GO TO 250
C
      IF(STEP.LT.19500)GO TO 71
   72 CONTINUE
      WRITE(10,73)T,AMP(1),AMP(2),AMP(3),AMP(4),AMP(5)
     1 ,AMP(6),AMP(7),AMP(8),AMP(9),AMP(10),AMP(11),
     1 AMP(12)
   73 FORMAT(13F8.4)
   71 CONTINUE
      DO 30 J=1,N
      ZDD(J)=((1.0/(ALPHA*(DELT**2)))*(Z(J)-ZI(J)))
     1 -((1.0/(ALPHA*DELT))*ZDI(J))
     1 -(((1.0/(2.0*ALPHA))-1.0)*ZDDI(J))
      ZD(J)=ZDI(J)+((1.0-BETA)*DELT*ZDDI(J))
     1 +(BETA*DELT*ZDD(J))
   30 CONTINUE
      DO 280 II=1,N
      ZI(II)=Z(II)
      ZDI(II)=ZD(II)
      ZDDI(II)=ZDD(II)
  280 CONTINUE
      GO TO 270
  250 CONTINUE
      DO 260 KK=1,N
      ZAI(KK)=0.5*(Z(KK)+ZA(KK))
  260 CONTINUE
```

```
      GO TO 200
 270  CONTINUE
      DO 2100 KK=1,N
2100  ZAI(KK)=Z(KK)
 100  CONTINUE
  43  CONTINUE
      STOP
      END
```

F.3.2 OPTIMIZATION PROGRAM - NON-LINEAR ROTOR SYSTEM

```
C       OPT12.FOR
C       THIS PROGRAM USES AN OPTIMIZATION ROUTINE TO FIND THE
C       EQUIVALENT LINEAR STIFNESS VALUES FOR A MDOF SYSTEM
C       WITH FORCED VIBRATION
C
C       12 DOF SYSTEM - ROTOR PROBLEM -
C
        DIMENSION X(4),XSTRT(4),RMAX(4),RMIN(4),PHI(8),
     1  PSI(1),W(1000)
        N=4
        NCONS=6
        NEQUS=0
        NPENAL=5
        DATA RMAX/5.0E07,5.0E07,5.0E07,5.0E07/
        DATA RMIN/1.0E07,1.0E07,1.0E07,1.0E07/
        DATA XSTRT/1.0E07,1.510625E07,
     1  1.0271875E07,1.418125E07/
        CALL SEEK(N,NCONS,NEQUS,NPENAL,RMAX,RMIN,XSTRT,X,U,
     1  PHI,PSI,NVIOL,W)
        CALL ANSWER(U,X,PHI,PSI,N,NCONS,NEQUS)
        STOP
        END
C
C       SUBROUTINE UREAL
C
        SUBROUTINE UREAL(X,U)
        DIMENSION X(1)
        REAL M(12,12),K(12,12),KINV(12,12),D(12,12)
        REAL M1,M2,M3,M4,M5,M6
        REAL K22,K23,K24,K25,K32,K33,K34,K35
        REAL K42,K43,K44,K45,K52,K53,K54,K55
        REAL INERTIA,EE,PI,DIA,ZETA
        REAL KZZ1,KYY1,KZZ2,KYY2
        REAL RPM1,RPM2,WMIN,WMAX,W,
        REAL F5,F6,F7,F8,f55,f66,f77,f88
        REAL A1,A2,A3,A4,A5,A6,A7,A8,L,L2,L3,CONST1,
     1  A22,A23,A24,A25
        REAL A32,A33,A34,A35,A42,A43,A44,A45,A52,A53,
     1  A54,A55,L4,L5
        REAL A(4,4),KS(4,4)
        REAL EVR(12),CC(12,12),CCCC(12,12)
        REAL SS(12),PHASE(12),FR(12),FI(12)
        REAL THETA2,THETA3,ALPHA1,ALPHA2,TTT,THETA4,THETA5
        REAL EC2,EC3,EC4,EC5,CONVERS
        COMPLEX DD(12,12),EV(12),ZV(12,12),VV(3,12)
        COMPLEX CCC(12,12),ZVI(12,12),ZVT(12,12),ZVTI(12,12),
        COMPLEX DAMP(12,12),PP(12,12),CHECK(4,4),CK(12,12)
        COMPLEX Z(12,12),ZINVS(12,12),F(12),AMP(12)
        COMPLEX FACTOR1,FACTOR2,FACTOR3,FACTOR4,number
```

```
      COMPLEX FACTOR5,FACTOR6,FACTOR7,FACTOR8,FACTOR9,
      COMPLEX FACTOR10,FACTOR11,FACTOR12
      INTEGER N,IND,NW,IER,KK
      DIMENSION WK(1000),XX(1000,12),TT(1000)
      DIMENSION UD(600),XNUM1(600),XNUM2(600),XNUM3(600),
      DIMENSION XNUM4(600),XNUM5(600),XNUM6(600),XNUM7(600)
      DIMENSION XNUM8(600),XNUM9(600),XNUM10(600)
      DIMENSION XNUM11(600),XNUM12(600)
      OPEN(UNIT=13,FILE='KKK.DAT',TYPE='OLD')
      OPEN(UNIT=16,FILE='MMM.DAT',TYPE='OLD')
      OPEN(UNIT=14,FILE='DAMP.DAT',TYPE='OLD')
      OPEN(UNIT=12,FILE='DAT.DAT',TYPE='OLD')
C
C     INPUT NUMERICAL SOLUTION DATA
C
      REWIND 12
      DATA=501
      DO 5 II=1,DATA
      READ(12,*)TT(II),XNUM1(II),XNUM2(II),XNUM3(II),
     1 XNUM4(II),XNUM5(II),XNUM6(II),XNUM7(II),XNUM8(II),
     1 XNUM9(II),XNUM10(II),XNUM11(II),XNUM12(II)
    5 CONTINUE
C
C     INPUT DATA
C
      N=12
      PI=3.14159
      EC2=0.00002
      EC3=0.00002
      EC4=0.00002
      EC5=0.00002
      THETA2=0.785
      THETA3=0.785
      THETA4=2.355
      THETA5=2.355
      M1=0.10
      M2=30.0
      M3=30.0
      M4=30.0
      M5=30.0
      M6=0.10
C
      KZZ1=X(1)
      KZZ2=X(2)
      KYY1=X(3)
      KYY2=X(4)
C
C     CALCULATE LINEARIZED TERMS OF STIFFNESS MATRIX
C

      K(1,1)=KZZ1+((A1**2)*K22)+(2*K23*A1*A3)+(2*K24*A1
     1 *A5)+(2*K25*A1*A7)+((A3**2)*K33)+(2*K34*A3*A5)
```

```
     1 +(2*K35*A3*A7)+((A5**2)*K44)+(2*K45*A5*A7)
     1 +((A7**2)*K55)
       K(2,2)=KYY1+((A1**2)*K22)+(2*K23*A1*A3)+(2*K24
     1 *A1*A5)+(2*K25*A1*A7)+(2*K34*A3*A5)+(2*K35*A3*A7)
     1 +((A5**2)*K44)+(2*K45*A5*A7)+((A7**2)*K55)
     1 +((A3**2)*K33)
       K(11,11)=KZZ2+((A2**2)*K22)+(2*A4*A2*K23)
     1 +(2*A2*A6*K24)+(2*A2*A8*K25)+((A4**2)*K33)
     1 +(2*A4*A6*K34)+(2*A4*A8*K35)+((A6**2)*K44)
     1 +(2*A6*A8*K45)+((A8**2)*K55)
       K(12,12)=KYY2+((A2**2)*K22)+(2*A4*A2*K23)
     1 +(2*A2*A6*K24)+(2*A2*A8*K25)+((A4**2)*K33)
     1 +(2*A4*A6*K34)+(2*A4*A8*K35)+((A6**2)*K44)
     1 +(2*A6*A8*K45)+((A8**2)*K55)
C
       RPM1=2550.0
       W=(RPM1*(2.0*PI))/60.0
C
C      INPUT FORCING FUNCTION
C
       FR(1)=0.0
       FR(2)=0.0
       FR(3)=M2*EC2*(W**2)*COS(THETA2)
       FR(4)=M2*EC2*(W**2)*COS(THETA2+((3*PI)/2))
       FR(5)=M3*EC3*(W**2)*COS(THETA3)
       FR(6)=m3*ec3*(W**2)*COS(THETA3+((3*pi)/2))
       FR(7)=m4*ec4*(W**2)*COS(THETA4)
       FR(8)=m4*ec4*(W**2)*COS(THETA4+((3*pi)/2))
       FR(9)=M5*EC5*(W**2)*COS(THETA5)
       FR(10)=M5*EC5*(W**2)*COS(THETA5+((3*PI)/2))
       FR(11)=0.0
       FR(12)=0.0
       FI(1)=0.0
       FI(2)=0.0
       FI(3)=M2*EC2*(W**2)*SIN(THETA2)
       FI(4)=M2*EC2*(W**2)*SIN(THETA2+((3*PI)/2))
       FI(5)=m3*ec3*(w**2)*sin(theta3)
       FI(6)=m3*ec3*(w**2)*sin(theta3+((3*pi)/2))
       FI(7)=m4*ec4*(w**2)*sin(theta4)
       FI(8)=m4*ec4*(w**2)*sin(theta4+((3*pi)/2))
       FI(9)=M5*EC5*(W**2)*SIN(THETA5)
       FI(10)=M5*EC5*(W**2)*SIN(THETA5+((3*PI)/2))
       FI(11)=0.0
       FI(12)=0.0
       F(1)=CMPLX(0.0,0.0)
       F(2)=CMPLX(0.0,0.0)
       F(3)=CMPLX(FR(3),FI(3))
       F(4)=CMPLX(FR(4),FI(4))
       F(5)=CMPLX(FR(5),FI(5))
       F(6)=CMPLX(FR(6),FI(6))
       F(7)=CMPLX(FR(7),FI(7))
       F(8)=CMPLX(FR(8),FI(8))
```

```
      F(9)=CMPLX(FR(9),FI(9))
      F(10)=CMPLX(FR(10),FI(10))
      F(11)=CMPLX(0.0,0.0)
      F(12)=CMPLX(0.0,0.0)
C
C    CALCULATION OF RESPONSE
C
      DO 44 I=1,N
      DO 44 J=1,N
   44 Z(I,J)=CMPLX(-M(I,J)*W**2+K(I,J),CCCC(I,J)*W)
      CALL LINCG(12,Z,12,ZINVS,12)
      DO 45 I=1,N
      AMP(I)=CMPLX(0.0,0.0)
      DO 46 J=1,N
   46 AMP(I)=AMP(I)+ZINVS(I,J)*F(J)
      IF(ABS(REAL(AMP(I))).LT.1.E-20)
      AMP(I)=CMPLX(1.E-20,AIMAG(AMP(I)))
      CONVERS=(1000.0/25.4)
      SS(I)=CABS(AMP(I))*CONVERS
   45 PHASE(I)=57.296*ATAN2(AIMAG(AMP(I)),REAL(AMP(I)))
      WRITE(6,92)W,(SS(I),I=1,N)
      WRITE(6,93)(PHASE(I),I=1,N)
   92 FORMAT(/,'  FREQ:     ',E12.4,/,'  AMP(I):        1
     ',6F8.4,/,10X,6F8.4)
   93 FORMAT('PHASE(I):',5E12.4,/,10X,5E12.4,/,10X,2E12.4)
C
C     STEADY STATE RESPONSE
C
      DO 305 I=1,DATA
      NUMBER=CMPLX(COS(W*TT(I)),SIN(W*TT(I)))
      FACTOR1=AMP(1)*NUMBER
      factor2=amp(2)*number
      FACTOR3=AMP(3)*NUMBER
      factor4=amp(4)*number
      FACTOR5=AMP(5)*NUMBER
      factor6=amp(6)*number
      FACTOR7=AMP(7)*NUMBER
      factor8=amp(8)*number
      FACTOR9=AMP(9)*NUMBER
      factor10=amp(10)*number
      FACTOR11=AMP(11)*NUMBER
      factor12=amp(12)*number
      XX(I,1)=REAL(FACTOR1)*CONVERS
      xx(i,2)=real(factor2)*convers
      XX(I,3)=REAL(FACTOR3)*CONVERS
      xx(i,4)=real(factor4)*convers
      XX(I,5)=REAL(FACTOR5)*CONVERS
      xx(i,6)=real(factor6)*convers
      XX(I,7)=REAL(FACTOR7)*CONVERS
      xx(i,8)=real(factor8)*convers
      XX(I,9)=REAL(FACTOR9)*CONVERS
      xx(i,10)=real(factor10)*convers
```

```
      XX(I,11)=REAL(FACTOR11)*CONVERS
      xx(i,12)=real(factor12)*convers
  305 CONTINUE
   15 CONTINUE
C
C     FIND VALUE OF THE OPTIMIZATION  FUNCTION - U -
C
      D O    4 0 0    I I = 1 , D A T A
      UD(II)=((XNUM3(II)-XX(II,3))**2)+((XNUM4(II)-XX(II,4))
     1 **2)+((XNUM5(II)-XX(II,5))**2)+((XNUM6(II)
     1 -XX(II,6))**2)+((XNUM7(II)-XX(II,7))**2)+((XNUM8(II)-
     1 XX(II,8))**2)+((XNUM9(II)-XX(II,9))**2)+((XNUM10(II)-
     1 XX(II,10))**2)
  400 CONTINUE
      UU=UD(1)
      DO 405 II=1,DATA
      JJ=II+1
      UU=UU+UD(JJ)
  405 CONTINUE
      U=UU*1.0E12
      WRITE(6,*)U
C
      END
C
C     SUBROUTINE CONST
C
      SUBROUTINE CONST(X,NCONS,PHI)
      DIMENSION X(1),PHI(1)
      PHI(1)=5.0E07-X(1)
      PHI(2)=X(1)-1.0E07
      PHI(3)=5.0E07-X(2)
      PHI(4)=X(2)-1.0E07
      PHI(5)=5.0E07-X(3)
      PHI(6)=X(3)-1.0E07
      PHI(7)=5.0E07-X(4)
      PHI(8)=X(4)-1.0E07
      RETURN
      END
C
C    SUBROUTINE EQUAL
C
      SUBROUTINE EQUAL(X,PSI,NEQUS)
      DIMENSION X(1),PSI(1)
      RETURN
      END
C
C     SUBROUTINE CMPLY
C
      SUBROUTINE CMPLY(A,B,C,N)
      COMPLEX A(N,N),B(N,N),C(N,N)
      DO 40 I=1,N
      DO 40 J=1,N
```

```
      C(I,J)=0
      DO 40 K=1,N
   40 C(I,J)=C(I,J)+A(I,K)*B(K,J)
      RETURN
      END
C
C     SUBROUTINE TRANS(A,B,N)
C     B=TRANSPOSE OF A
      SUBROUTINE TRANS(A,B,N)
      COMPLEX A(N,N),B(N,N)
      DO 40 I=1,N
      DO 40 J=1,N
      B(I,J)=A(J,I)
   40 CONTINUE
      RETURN
      END
```

F.3.3 BALANCING PROGRAM

```
C       BAL7.FOR
C       -  12 DOF SYSTEM
C       -  CALCULATES THE UNBALANCE RESPONSE OF A LINEAR ROTOR
C       -  DETERMINES THE BALANCE WEIGHTS USING THE
C          LEAST SQUARES INFLUENCE COEFFICIENT METHOD..
        REAL M(12,12),K(12,12),KINV(12,12),D(12,12)
        REALM1,M2,M3,M4,M5,M6,K22,K23,K24,K25,K32,K33,K34,K35
        REAL K42,K43,K44,K45,K52,K53,K54,K55
        REAL INERTIA,EE,PI,DIA,ZETA
        REAL KZZ1,KYY1,KZZ2,KYY2
        REAL RPM(3),W,F5,F6,F7,F8,f55,f66,f77,f88
        REAL A1,A2,A3,A4,A5,A6,A7,A8,L,L2,L3,L4,L5,CONST1
        REAL A22,A23,A24,A25,A32,A33,A34,A35
        REAL A42,A43,A44,A45,A52,A53,A54,A55
        REAL A(4,4),KS(4,4)
        REAL WK(1000)
        REAL EVR(12),CC(12,12),CCCC(12,12)
        REAL AMP(12),PHASE(12),FR(5,12),FI(5,12)
        REAL THETA2,THETA3,TTT,THETA4,THETA5
        REAL ALPHA1,ALPHA2,ALPHA3
        REAL EC2,EC3,EC4,EC5
        REAL WT1,WT2,WT3,ET1,ET2,ET3
        REAL T1X,T1Y,T2X,T2Y,T3X,T3Y
        REAL BAL1,BAL2,BAL3
        REAL GAMMA1,GAMMA2,GAMMA3
        REAL COR11,COR21,COR31,COR12,COR22,COR32
        REAL COR13,COR23,COR33
        REAL ANG11,ANG21,ANG31,ANG12,ANG22,ANG32
        REAL ANG13,ANG23,ANG33
        REAL R(4),DELT,CALC(12,12),INPUT(12,12)
        REAL AMP1(12),PHASE1(12)
        COMPLEX DD(12,12),EV(12),ZV(12,12),VV(5,12)
        COMPLEX CCC(12,12),ZVI(12,12),ZVT(12,12)
        COMPLEX ZVTI(12,12),CK(12,12),NUMBER
        COMPLEX DAMP(12,12),PP(12,12),CHECK(4,4)
        COMPLEXZ(12,12),ZINVS(12,12),X(12),F(12),AA(12)
        COMPLEX X10,X20,X30,X40,X50,X60
        COMPLEX X11,X21,X31,X41,X51,X61
        COMPLEX X12,X22,X32,X42,X52,X62
        COMPLEX X13,X23,X33,X43,X53,X63
        COMPLEX XX(6),T1,T2,T3
        COMPLEX ALPH(6,3),ALCJ(6,3),ALCJT(3,6),PROD(3,3)
        COMPLEX PRODI(3,3),ALPH2(3,6),CW(3),ACW(6,1),XR(6,1)
        INTEGER N,IND,NW,IER,KK
        OPEN(UNIT=10,FILE='BAL7.DAT',TYPE='NEW')
        OPEN(UNIT=14,FILE='DAMP.DAT',TYPE='OLD')
C
C       INPUT DATA
C
```

```
      N=12
      PI=3.14159
      DIA=0.076
      RAD=DIA/2.0
      INERTIA=PI*((RAD**4.0)/4.0)
      EE=2E11
      ZETA1=0.02
      ZETA2=0.02
      ZETA3=0.02
      ZETA4=0.02
      ZETA5=0.02
      ZETA6=0.02
      ZETA7=0.02
      ZETA8=0.02
      ZETA9=0.002
      ZETA10=0.002
      ZETA11=0.05
      ZETA12=0.05
      EC2=0.00002
      EC3=0.00002
      EC4=0.00002
      EC5=0.00002
      THETA2=0.785
      THETA3=0.785
      THETA4=2.355
      THETA5=2.355
      KZZ1=1.2121177e07
      KYY1=1.2109146e07
      KZZ2=1.0650005e07
      KYY2=1.0650005e07
      L=1.20
      L2=.20
      L3=.45
      L4=.75
      L5=1.0
      M1=0.05
      M2=20.0
      M3=20.0
      M4=20.0
      M5=20.0
      M6=0.05
C
      A1=1-L2/L
      A2=L2/L
      A3=1-(L3/L)
      A4=L3/L
      A5=1-(L4/L)
      A6=L4/L
      A7=1-(L5/L)
      A8=L5/L
      CONST1=-(1.0/(EE*INERTIA))
      A22=CONST1*((((L-L2)*(L2**3))/(6.0*L))+(((L-L2)*L2
```

```
     1 *((((L-L2)**2)*L)-(L**3)))/(6.0*(L**2))))
       A33=CONST1*((((L-L3)*(L3**3))/(6.0*L))+(((L-L3)*L3
     1 *((((L-L3)**2)*L)-(L**3)))/(6.0*(L**2))))
       A44=CONST1*((((L-L4)*(L4**3))/(6.0*L))+(((L-L4)*L4
     1 *((((L-L4)**2)*L)-(L**3)))/(6.0*(L**2))))
       A55=CONST1*((((L-L5)*(L5**3))/(6.0*L))+(((L-L5)*L5
     1 *((((L-L5)**2)*L)-(L**3)))/(6.0*(L**2))))
       A23=CONST1*((((L-L2)*(L3**3))/(6.0*L))-(((L3-L2)
     1 **3)/6)+(((L-L2)*L3*((((L-L2)**2)*L)-(L**3)))/(6.0
     1 *(L**2))))
       A42=CONST1*((((L-L2)*(L4**3))/(6.0*L))-(((L4-L2)
     1 **3)/6)+(((L-L2)*L4*((((L-L2)**2)*L)-(L**3)))/(6.0
     1 *(L**2))))
       A52=CONST1*((((L-L2)*(L5**3))/(6.0*L))-(((L5-L2)
     1 **3)/6)+(((L-L2)*L5*((((L-L2)**2)*L)-(L**3)))/(6.0
     1 *(L**2))))
       A43=CONST1*((((L-L3)*(L4**3))/(6.0*L))-(((L4-L3)
     1 **3)/6)+(((L-L3)*L4*((((L-L3)**2)*L)-(L**3)))/(6.0
     1 *(L**2))))
       A53=CONST1*((((L-L3)*(L5**3))/(6.0*L))
     1 -(((L5-L3)**3)/6)+(((L-L3)*L5*((((L-L3)**2)*L)
     1 -(L**3)))/(6.0*(L**2))))
       A54=CONST1*((((L-L4)*(L5**3))/(6.0*L))-(((L5-L4)
     1 **3)/6)+(((L-L4)*L5*((((L-L4)**2)*L)-(L**3)))/(6.0
     1 *(L**2))))
       A32=A23
       A24=A42
       A34=A43
       A25=A52
       A35=A53
       A45=A54
       A(1,1)=A22
       A(1,2)=A23
       A(1,3)=A24
       A(1,4)=A25
       A(2,1)=A32
       A(2,2)=A33
       A(2,3)=A34
       A(2,4)=A35
       A(3,1)=A42
       A(3,2)=A43
       A(3,3)=A44
       A(3,4)=A45
       A(4,1)=A52
       A(4,2)=A53
       A(4,3)=A54
       A(4,4)=A55
       DO 23 J=1,4
    23 WRITE(10,*)A(J,1),A(J,2),A(J,3),A(J,4)
C
C      FIND [KS]=INVERSE OF [A]
C
```

```
      CALL LINDS(4,A,4,KS,4)
      K22=KS(1,1)
      K23=KS(1,2)
      K24=KS(1,3)
      K25=KS(1,4)
      K32=KS(2,1)
      K33=KS(2,2)
      K34=KS(2,3)
      K35=KS(2,4)
      K42=KS(3,1)
      K43=KS(3,2)
      K44=KS(3,3)
      K45=KS(3,4)
      K52=KS(4,1)
      K53=KS(4,2)
      K54=KS(4,3)
      K55=KS(4,4)
C
C    INPUT MASS MATRIX
C
      DO 50 I=1,12
      DO 50 J=1,12
      M(I,J)=0.0
   50 CONTINUE
      M(1,1)=M1
      M(2,2)=M1
      M(3,3)=M2
      M(4,4)=M2
      M(5,5)=M3
      M(6,6)=M3
      M(7,7)=M4
      M(8,8)=M4
      M(9,9)=M5
      M(10,10)=M5
      M(11,11)=M6
      M(12,12)=M6
C
C    INPUT LINEAR SYSTEM STIFNESS MATRIX
C
      K(1,1)=KZZ1+((A1**2)*K22)+(2*K23*A1*A3)+(2*K24*A1
     1 *A5)+(2*K25*A1*A7)+((A3**2)*K33)+(2*K34*A3*A5)+(2
     1 *K35*A3*A7)+((A5**2)*K44)+(2*K45*A5*A7)
     1 +((A7**2)*K55)
      K(1,2)=0.0
      K(1,3)=(-A1*K22)-(A3*K23)-(A5*K24)-(A7*K25)
      K(1,4)=0.0
      K(1,5)=(-A1*K23)-(A3*K33)-(A5*K34)-(A7*K35)
      K(1,6)=0.0
      K(1,7)=(-A1*K24)-(A3*K34)-(A5*K44)-(A7*K45)
      K(1,8)=0.0
      K(1,9)=(-A1*K25)-(A3*K35)-(A5*K45)-(A7*K55)
      K(1,10)=0.0
```

```
      K(1,11)=(A1*A2*K22)+(A1*A4*K23)+(A2*A3*K23)
     1 +(A2*A5*K24)+(A1*A6*K24)+(A2*A7*K25)+(A1*A8*K25)
     1 +(A3*A4*K33)+(A3*A6*K34)+(A4*A5*K34)+(A4*A7*K35)
     1 +(A3*A8*K3)+(A5*A6*K44)+(A6*A7*K45)+(A5*A8*K45)
     1 +(A7*A8*K55)
      K(1,12)=0.0
      K(2,1)=0.0
      K(2,2)=KYY1+((A1**2)*K22)+(2*K23*A1*A3)+(2*K24
     1 *A1*A5)+(2*K25*A1*A7)+(2*K34*A3*A5)+(2*K35
     1 *A3*A7)+((A5**2)*K44)+(2*K45*A5*A7)
     1 +((A7**2)*K55)+((A3**2)*K33)
      K(2,3)=0.0
      K(2,4)=(-A1*K22)-(A3*K23)-(A5*K24)-(A7*K25)
      K(2,5)=0.0
      K(2,6)=(-A1*K23)-(A3*K33)-(A5*K34)-(A7*K35)
      K(2,7)=0.0
      K(2,8)=(-A1*K24)-(A3*K34)-(A5*K44)-(A7*K45)
      K(2,9)=0.0
      K(2,10)=(-A1*K25)-(A3*K35)-(A5*K45)-(A7*K55)
      K(2,11)=0.0
      K(2,12)=(A1*A2*K22)+(A1*A4*K23)+(A2*A3*K23)
     1 +(A2*A5*K24)+(A1*A6*K24)+(A2*A7*K25)+(A1*A8*K25)
     1 +(A3*A4*K33)+(A3*A6*K34)+(A4*A5*K34)+(A4*A7*K35)
     1 +(A3*A8*K35)+(A5*A6*K44)+(A6*A7*K45)+(A5*A8*K45)
     1 +(A7*A8*K55)
      K(3,1)=(-A1*K22)-(A3*K23)-(A5*K24)-(A7*K25)
      K(3,2)=0.0
      K(3,3)=K22
      K(3,4)=0.0
      K(3,5)=K23
      K(3,6)=0.0
      K(3,7)=K24
      K(3,8)=0.0
      K(3,9)=K25
      K(3,10)=0.0
      K(3,11)=(-A2*K22)-(A4*K23)-(A6*K24)-(A8*K25)
      K(3,12)=0.0
      K(4,1)=0.0
      K(4,2)=(-A1*K22)-(A3*K23)-(A5*K24)-(A7*K25)
      K(4,3)=0.0
      K(4,4)=K22
      K(4,5)=0.0
      K(4,6)=K23
      K(4,7)=0.0
      K(4,8)=K24
      K(4,9)=0.0
      K(4,10)=K25
      K(4,11)=0.0
      K(4,12)=(-A2*K22)-(A4*K23)-(A6*K24)-(A8*K25)
      K(5,1)=(-A1*K23)-(A3*K33)-(A5*K34)-(A7*K35)
      K(5,2)=0.0
      K(5,3)=K23
```

```
K(5,4)=0.0
K(5,5)=K33
K(5,6)=0.0
K(5,7)=K34
K(5,8)=0.0
K(5,9)=K35
K(5,10)=0.0
K(5,11)=(-A2*K23)-(A4*K33)-(A6*K34)-(A8*K35)
K(5,12)=0.0
K(6,1)=0.0
K(6,2)=(-A1*K23)-(A3*K33)-(A5*K34)-(A7*K35)
K(6,3)=0.0
K(6,4)=K23
K(6,5)=0.0
K(6,6)=K33
K(6,7)=0.0
K(6,8)=K34
K(6,9)=0.0
K(6,10)=K35
K(6,11)=0.0
K(6,12)=(-A2*K23)-(A4*K33)-(A6*K34)-(A8*K35)
K(7,1)=(-A1*K24)-(A3*K34)-(A5*K44)-(A7*K45)
K(7,2)=0.0
K(7,3)=K24
K(7,4)=0.0
K(7,5)=K34
K(7,6)=0.0
K(7,7)=K44
K(7,8)=0.0
K(7,9)=K45
K(7,10)=0.0
K(7,11)=(-A2*K24)-(A4*K34)-(A6*K44)-(A8*K45)
K(7,12)=0.0
K(8,1)=0.0
K(8,2)=(-A1*K24)-(A3*K34)-(A5*K44)-(A7*K45)
K(8,3)=0.0
K(8,4)=K24
K(8,5)=0.0
K(8,6)=K34
K(8,7)=0.0
K(8,8)=K44
K(8,9)=0.0
K(8,10)=K45
K(8,11)=0.0
K(8,12)=(-A2*K24)-(A4*K34)-(A6*K44)-(A8*K45)
K(9,1)=(-A1*K25)-(A3*K35)-(A5*K45)-(A7*K55)
K(9,2)=0.0
K(9,3)=K25
K(9,4)=0.0
K(9,5)=K35
K(9,6)=0.0
K(9,7)=K45
```

```
      K(9,8)=0.0
      K(9,9)=K55
      K(9,10)=0.0
      K(9,11)=(-A2*K25)-(A4*K35)-(A6*K45)-(A8*K55)
      K(10,1)=0.0
      K(10,2)=(-A1*K25)-(A3*K35)-(A5*K45)-(A7*K55)
      K(10,3)=0.0
      K(10,4)=K25
      K(10,5)=0.0
      K(10,6)=K35
      K(10,7)=0.0
      K(10,8)=K45
      K(10,9)=0.0
      K(10,10)=K55
      K(10,11)=0.0
      K(10,12)=(-A2*K25)-(A4*K35)-(A6*K45)-(A8*K55)
      K(11,1)=(A1*A2*K22)+(A2*A3*K23)+(A1*A4*K23)
     1 +(A2*A5*K24)+(A1*A6*K24)+(A2*A7*K25)+(A1*A8*K25)
     1 +(A3*A4*K33)+(A3*A6*K34)+(A4*A5*K34)+(A4*A7*K35)
     1 +(A3*A8*K35)+(A5*A6*K44)+(A6*A7*K45)+(A5*A8*K45)
     1 +(A7*A8*K55)
      K(11,2)=0.0
      K(11,3)=(-A2*K22)-(A4*K23)-(A6*K24)-(A8*K25)
      K(11,4)=0.0
      K(11,5)=(-A2*K23)-(A4*K33)-(A6*K34)-(A8*K35)
      K(11,6)=0.0
      K(11,7)=(-A2*K24)-(A4*K34)-(A6*K44)-(A8*K45)
      K(11,8)=0.0
      K(11,9)=(-A2*K25)-(A4*K35)-(A6*K45)-(A8*K55)
      K(11,10)=0.0
      K(11,11)=KZZ2+((A2**2)*K22)+(2*A4*A2*K23)
     1 +(2*A2*A6*K24)+(2*A2*A8*K25)+((A4**2)*K33)
     1 +(2*A4*A6*K34)+(2*A4*A8*K35)+((A6**2)*K44)
     1 +(2*A6*A8*K45)+((A8**2)*K55)
      K(11,12)=0.0
      K(12,1)=0.0
      K(12,2)=(A1*A2*K22)+(A2*A3*K23)+(A1*A4*K23)
     1 +(A2*A5*K24)+(A1*A6*K24)+(A2*A7*K25)+(A1*A8*K25)
     1 +(A3*A4*K33)+(A3*A5*K34)+(A4*A5*K34)+(A4*A7*K35)
     1 +(A3*A8*K35)+(A5*A6*K44)+(A6*A7*K45)+(A5*A8*K45)
     1 +(A7*A8*K55)
      K(12,3)=0.0
      K(12,4)=(-A2*K22)-(A4*K23)-(A6*K24)-(A8*K25)
      K(12,5)=0.0
      K(12,6)=(-A2*K23)-(A4*K33)-(A6*K34)-(A8*K35)
      K(12,7)=0.0
      K(12,8)=(-A2*K24)-(A4*K34)-(A6*K44)-(A8*K45)
      K(12,9)=0.0
      K(12,10)=(-A2*K25)-(A4*K35)-(A6*K45)-(A8*K55)
      K(12,11)=0.0
      K(12,12)=KYY2+((A2**2)*K22)+(2*A4*A2*K23)
     1 +(2*A2*A6*K24)+(2*A2*A8*K25)+((A4**2)*K33)
```

```
     1 +(2*A4*A6*K34)+(2*A4*A8*K35)+((A6**2)*K44)
     1 +(2*A6*A8*K45)+((A8**2)*K55)
C
       CALL LINDS(12,K,12,KINV,12)
       CALL MRRRR(12,12,KINV,12,12,12,M,12,12,12,D,12)
       DO 30 I=1,12
       DO 30 J=1,12
       DD(I,J)=CMPLX(D(I,J))
   30  CONTINUE
       CALL EVCCG(12,DD,12,EV,ZV,12)
       DO 35 I=1,12
       EVR(I)=REAL(EV(I))
       EVR(I)=1./SQRT(EVR(I))
   35  CONTINUE
       WRITE(10,40)
   40  FORMAT(/,' THE EIGENVALUES ARE AS FOLLOWS:    ')
       WRITE(10,*)EVR(1),EVR(2),EVR(3),EVR(4),EVR(5),EVR(6),
     1 EVR(7),EVR(8),EVR(9),EVR(10),EVR(11),EVR(12)
       WRITE(10,36)
   36  FORMAT(/,' THE EIGENVECTORS ARE')
       DO 37 I=1,12
       WRITE(10,*)I
       DO 37 J=1,12
       WRITE(10,*)ZV(J,I)
   37  CONTINUE
C
C      CALCULATE DAMPING MATRIX FOR LINEAR SYSTEM
C
       CC(1,1)=ZETA1*EVR(12)*M(1,1)*2.0
       CC(2,2)=ZETA2*EVR(11)*M(2,2)*2.0
       CC(3,3)=ZETA3*EVR(10)*M(3,3)*2.0
       CC(4,4)=ZETA4*EVR(9)*M(4,4)*2.0
       CC(5,5)=ZETA5*EVR(8)*M(5,5)*2.0
       CC(6,6)=ZETA6*EVR(7)*M(6,6)*2.0
       CC(7,7)=ZETA7*EVR(6)*M(7,7)*2.0
       CC(8,8)=ZETA8*EVR(5)*M(8,8)*2.0
       CC(9,9)=ZETA9*EVR(4)*M(9,9)*2.0
       CC(10,10)=ZETA10*EVR(3)*M(10,10)*2.0
       CC(11,11)=ZETA11*EVR(2)*M(11,11)*2.0
       CC(12,12)=ZETA12*EVR(1)*M(12,12)*2.0
       DO 900 I=1,12
       DO 900 J=1,12
       CCC(I,J)=CMPLX(CC(I,J),0.0)
  900  CONTINUE
       CALL LINCG(12,ZV,12,ZVI,12)
       CALL TRANS(ZV,ZVT,12)
       CALL LINCG(12,ZVT,12,ZVTI,12)
       CALL CMPLY(ZVTI,CCC,PP,12)
       CALL CMPLY(PP,ZVI,DAMP,12)
C
C      READ IN DAMPING MATRIX FOR LINEAR SYSTEM
C
```

```
      DO 902 I=1,12
      DO 902 J=1,12
      READ(14,*)INPUT(i,j)
      CALC(I,J)=REAL(DAMP(I,J))
C
C     SELECT DAMPING MATRIX
C
C     CCCC(i,j)=CALC(i,j)
      CCCC(i,j)=INPUT(i,j)
  902 CONTINUE
      REWIND 14
C
C     BALANCING SPEEDS:RUN 1: W1
C                      RUN 2 : W2
C                      RUN 3 : W3
C
      RPM(1)=EVR(11)
      RPM(2)=EVR(9)
      RPM(3)=EVR(7)
C
C     INPUT FORCING FUNCTION
C
      DO 43 LL=1,3
      W=RPM(LL)
      WRITE(10,*)W
C
C     INPUT TRIAL WEIGHT VALUES
C
      WT1=0.006
      WT2=0.006
      WT3=0.006
      ET1=.152
      ET2=.152
      ET3=.152
      ALPHA1=0.785
      ALPHA2=4.713
      ALPHA3=2.356
C
C     INPUT ANY CORRECTION WEIGHTS
C
      COR11=0.0
      COR21=0.0
      COR31=0.0
      ANG11=-2.608977
      ANG21=-1.746057
      ANG31=-0.740426
      COR12=0.0
      COR22=0.0
      COR32=0.0
      ANG12=0.0
      ANG22=0.0
      ANG32=0.0
```

```
      COR13=0.0
      COR23=0.0
      COR33=0.0
      ANG13=0.0
      ANG23=0.0
      ANG33=0.0
C
C     ORIGINAL UNBALANCE FORCES- PLANES 2,3,4, AND 5
C     + ANY CORRECTION WEIGTHS - PLANES 2,3,AND 5
C
      FR(1,1)=0.0
      FR(1,2)=0.0
      FR(1,3)=M2*EC2*(W**2)*COS(THETA2)
    1 +(COR11*ET1*(W**2)*COS(ANG11))
    1 +(COR12*ET1*(W**2)*COS(ANG12))
    1 +(COR13*ET1*(W**2)*COS(ANG13))
      FR(1,4)=M2*EC2*(W**2)*COS(THETA2+((3*PI)/2))
    1 +(COR11*ET1*(W**2)*COS(ANG11+((3*PI)/2)))
    1 +(COR12*ET1*(W**2)*COS(ANG12+((3*PI)/2)))
    1 +(COR13*ET1*(W**2)*COS(ANG13+((3*PI)/2)))
      FR(1,5)=m3*ec3*(w**2)*cos(theta3)
    1 +(COR21*ET2*(W**2)*COS(ANG21))
    1 +(COR22*ET2*(W**2)*COS(ANG22))
    1 +(COR23*ET2*(W**2)*COS(ANG23))
      FR(1,6)=m3*ec3*(w**2)*cos(theta3+((3*pi)/2))
    1 +(COR21*ET2*(W**2)*COS(ANG21+((3*PI)/2)))
    1 +(COR22*ET2*(W**2)*COS(ANG22+((3*PI)/2)))
    1 +(COR23*ET2*(W**2)*COS(ANG23+((3*PI)/2)))
      FR(1,7)=m4*ec4*(w**2)*cos(theta4)
      FR(1,8)=m4*ec4*(w**2)*cos(theta4+((3*pi)/2))
      FR(1,9)=M5*EC5*(W**2)*COS(THETA5)
    1 +(COR31*ET3*(W**2)*COS(ANG31))
    1 +(COR32*ET3*(W**2)*COS(ANG32))
    1 +(COR33*ET3*(W**2)*COS(ANG33))
      FR(1,10)=M5*EC5*(W**2)*COS(THETA5+((3*PI)/2))
    1 +(COR31*ET3*(W**2)*COS(ANG31+((3*PI)/2)))
    1 +(COR32*ET3*(W**2)*COS(ANG32+((3*PI)/2)))
    1 +(COR33*ET3*(W**2)*COS(ANG33+((3*PI)/2)))
      FR(1,11)=0.0
      FR(1,12)=0.0
      FI(1,1)=0.0
      FI(1,2)=0.0
      FI(1,3)=M2*EC2*(W**2)*SIN(THETA2)
    1 +(COR11*ET1*(W**2)*SIN(ANG11))
    1 +(COR12*ET1*(W**2)*SIN(ANG12))
    1 +(COR13*ET1*(W**2)*SIN(ANG13))
      FI(1,4)=M2*EC2*(W**2)*SIN(THETA2+((3*PI)/2))
    1 +(COR11*ET1*(W**2)*SIN(ANG11+((3*PI)/2)))
    1 +(COR12*ET1*(W**2)*SIN(ANG12+((3*PI)/2)))
    1 +(COR13*ET1*(W**2)*SIN(ANG13+((3*PI)/2)))
      FI(1,5)=m3*ec3*(w**2)*sin(theta3)
    1 +(COR21*ET2*(W**2)*SIN(ANG21))
```

```
     1 +(COR22*ET2*(W**2)*SIN(ANG22))
     1 +(COR23*ET2*(W**2)*SIN(ANG23))
       FI(1,6)=m3*ec3*(w**2)*sin(theta3+((3*pi)/2))
     1 +(COR21*ET2*(W**2)*SIN(ANG21+((3*PI)/2)))
     1 +(COR22*ET2*(W**2)*SIN(ANG22+((3*PI)/2)))
     1 +(COR23*ET2*(W**2)*SIN(ANG23+((3*PI)/2)))
       FI(1,7)=m4*ec4*(w**2)*sin(theta4)
       FI(1,8)=m4*ec4*(w**2)*sin(theta4+((3*pi)/2))
       FI(1,9)=M5*EC5*(W**2)*SIN(THETA5)
     1 +(COR31*ET3*(W**2)*SIN(ANG31))
     1 +(COR32*ET3*(W**2)*SIN(ANG32))
     1 +(COR33*ET3*(W**2)*SIN(ANG33))
       FI(1,10)=M5*EC5*(W**2)*SIN(THETA5+((3*PI)/2))
     1 +(COR31*ET3*(W**2)*SIN(ANG31+((3*PI)/2)))
     1 +(COR32*ET3*(W**2)*SIN(ANG32+((3*PI)/2)))
     1 +(COR33*ET3*(W**2)*SIN(ANG33+((3*PI)/2)))
       FI(1,11)=0.0
       FI(1,12)=0.0
C
C      UNBALANCE FORCE FROM TRAIL WEIGHT NO. 1 - PLANE 2
C
       FR(2,1)=FR(1,1)
       FR(2,2)=FR(1,2)
       FR(2,3)=FR(1,3)+(WT1*ET1*(W**2)*COS(ALPHA1))
       FR(2,4)=FR(1,4)+(WT1*ET1*(W**2)*COS(ALPHA1
     1 +((3*pi)/2)))
       FR(2,5)=FR(1,5)
       FR(2,6)=FR(1,6)
       FR(2,7)=FR(1,7)
       FR(2,8)=FR(1,8)
       FR(2,9)=FR(1,9)
       FR(2,10)=FR(1,10)
       FR(2,11)=FR(1,11)
       FR(2,12)=FR(1,12)
       FI(2,1)=FI(1,1)
       FI(2,2)=FI(1,2)
       FI(2,3)=FI(1,3)+(WT1*ET1*(W**2)*SIN(ALPHA1))
       FI(2,4)=FI(1,4)+(WT1*ET1*(W**2)*SIN(ALPHA1
     1 +((3*pi)/2)))
       FI(2,5)=FI(1,5)
       FI(2,6)=FI(1,6)
       FI(2,7)=FI(1,7)
       FI(2,8)=FI(1,8)
       FI(2,9)=FI(1,9)
       FI(2,10)=FI(1,10)
       FI(2,11)=FI(1,11)
       FI(2,12)=FI(1,12)
C
C      UNBALANCE FORCE FROM TRAIL WEIGHT NO. 2 - PLANE 3
C
       FR(3,1)=FR(1,1)
       FR(3,2)=FR(1,2)
```

```
      FR(3,3)=FR(1,3)
      FR(3,4)=FR(1,4)
      FR(3,5)=FR(1,5)+(WT2*ET2*(W**2)*COS(ALPHA2))
      FR(3,6)=FR(1,6)+(WT2*ET2*(W**2)*COS(ALPHA2
     1 +((3*pi)/2)))
      FR(3,7)=FR(1,7)
      FR(3,8)=FR(1,8)
      FR(3,9)=FR(1,9)
      FR(3,10)=FR(1,10)
      FR(3,11)=FR(1,11)
      FR(3,12)=FR(1,12)
      FI(3,1)=FI(1,1)
      FI(3,2)=FI(1,2)
      FI(3,3)=FI(1,3)
      FI(3,4)=FI(1,4)
      FI(3,5)=FI(1,5)+(WT2*ET2*(W**2)*SIN(ALPHA2))
      FI(3,6)=FI(1,6)+(WT2*ET2*(W**2)*SIN(ALPHA2
     1 +((3*pi)/2)))
      FI(3,7)=FI(1,7)
      FI(3,8)=FI(1,8)
      FI(3,9)=FI(1,9)
      FI(3,10)=FI(1,10)
      FI(3,11)=FI(1,11)
      FI(3,12)=FI(1,12)
C
C     UNBALANCE FORCE FROM THE  TRAIL WEIGHT NO. 3 - PLANE 5
C
      FR(4,1)=FR(1,1)
      FR(4,2)=FR(1,2)
      FR(4,3)=FR(1,3)
      FR(4,4)=FR(1,4)
      FR(4,5)=FR(1,5)
      FR(4,6)=FR(1,6)
      FR(4,7)=FR(1,7)
      FR(4,8)=FR(1,8)
      FR(4,9)=FR(1,9)+(WT3*ET3*(W**2)*COS(ALPHA3))
      FR(4,10)=FR(1,10)+(WT3*ET3*(W**2)*COS(ALPHA3
     1 +((3*pi)/2)))
      FR(4,11)=FR(1,11)
      FR(4,12)=FR(1,12)
      FI(4,1)=FI(1,1)
      FI(4,2)=FI(1,2)
      FI(4,3)=FI(1,3)
      FI(4,4)=FI(1,4)
      FI(4,5)=FI(1,5)
      FI(4,6)=FI(1,6)
      FI(4,7)=FI(1,7)
      FI(4,8)=FI(1,8)
      FI(4,9)=FI(1,9)+(WT3*ET3*(W**2)*SIN(ALPHA3))
      FI(4,10)=FI(1,10)+(WT3*ET3*(W**2)*SIN(ALPHA3
     1 +((3*pi)/2)))
      FI(4,11)=FI(1,11)
```

```
      FI(4,12)=FI(1,12)
C
C     FORCE VECTOR FOR THE IMEDANCE METHOD CALCULATION
C
      DO 2100 JJ=1,4
      WRITE(10,1400)JJ
 1400 FORMAT(/,' CASE  NO.',I8)
      F(1)=CMPLX(FR(JJ,1),FI(JJ,1))
      F(2)=CMPLX(FR(JJ,2),FI(JJ,2))
      F(3)=CMPLX(FR(JJ,3),FI(JJ,3))
      F(4)=CMPLX(FR(JJ,4),FI(JJ,4))
      F(5)=CMPLX(FR(JJ,5),FI(JJ,5))
      F(6)=CMPLX(FR(JJ,6),FI(JJ,6))
      F(7)=CMPLX(FR(JJ,7),FI(JJ,7))
      F(8)=CMPLX(FR(JJ,8),FI(JJ,8))
      F(9)=CMPLX(FR(JJ,9),FI(JJ,9))
      F(10)=CMPLX(FR(JJ,10),FI(JJ,10))
      F(11)=CMPLX(FR(JJ,11),FI(JJ,11))
      F(12)=CMPLX(FR(JJ,12),FI(JJ,12))
      WRITE(10,1299)
 1299 FORMAT(/,'   THE FORCE VECTOR IS:')
      DO 1300 I=1,N
 1300 WRITE(10,*)REAL(F(I)),AIMAG(F(I))
C
C     CALCULATION OF RESPONSE
C
      DO 44 I=1,N
      DO 44 J=1,N
   44 Z(I,J)=CMPLX(-M(I,J)*W**2+K(I,J),CCCC(I,J)*W)
      CALL LINCG(12,Z,12,ZINVS,12)
      DO 45 I=1,N
      X(I)=CMPLX(0.0,0.0)
      DO 46 J=1,N
   46 X(I)=X(I)+ZINVS(I,J)*F(J)
      I F ( A B S ( R E A L ( X ( I ) ) ) . L T . 1 . E - 2 0 )
      X(I)=CMPLX(1.E-20,AIMAG(X(I)))
      AMP(I)=CABS(X(I))*(1000.0/25.4)
      AMP1(I)=CABS(X(I))
      PHASE(I)=57.296*ATAN2(AIMAG(X(I)),REAL(X(I)))
   45 PHASE1(I)=PHASE(I)/(57.296)
      WRITE(10,700)
  700 FORMAT(/,' The complex ss amplitudes ( X(II) )are
     1 as follows:')
      DO 702 I=1,N
      WRITE(10,*)REAL(X(I)),AIMAG(X(I)),AMP1(I),PHASE1(I)
      WRITE(10,*)AMP(I),PHASE(I)
  702 CONTINUE
      WRITE(10,704)
  704 FORMAT(/,' The ss amplitudes are as follows:')
      WRITE(10,92)W,(AMP(I),I=1,N)
      WRITE(10,93)(PHASE(I),I=1,N)
   92 FORMAT(/,'  FREQ:            ',E12.4,/,'     AMP(I):
```

```
     1 ',5E12.4,/,10X,5E12.4,/,10X,5E12.4)
  93  FORMAT('  PHASE(I):',5E12.4,/10X,5E12.4,/,10X,5E12.4)
      DO 2000 II=1,12
      VV(JJ,II)=x(ii)
 2000 CONTINUE
 2100 CONTINUE
C
C     CALCULATE BALANCE WEIGHTS USING RESPONSE DATA
C     XIJ....I = MEASUREMENT PLANE
C            J = 0  ORIGINAL UNBALANCE
C            J = 1  FIRST TRIAL WEIGHT-PLANE 2
C            J = 2  SECOND TRIAL WEIGHT-PLANE 3
C            J = 3  THIRD TRIAL WEIGHT-PLANE 5
C
C     USING VERTICAL DISPLACEMENTS AT PLANES 1,2,3,4,5 AND 6
C
      X10=VV(1,1)
      X20=VV(1,3)
      X30=VV(1,5)
      X40=VV(1,7)
      X50=VV(1,9)
      X60=VV(1,11)
      X11=VV(2,1)
      X21=VV(2,3)
      X31=VV(2,5)
      X41=VV(2,7)
      X51=VV(2,9)
      X61=VV(2,11)
      X12=VV(3,1)
      X22=VV(3,3)
      X32=VV(3,5)
      X42=VV(3,7)
      X52=VV(3,9)
      X62=VV(3,11)
      X13=VV(4,1)
      X23=VV(4,3)
      X33=VV(4,5)
      X43=VV(4,7)
      X53=VV(4,9)
      X63=VV(4,11)
      XX(1)=X10
      XX(2)=X20
      XX(3)=X30
      XX(4)=X40
      XX(5)=X50
      XX(6)=X60
      WRITE(10,708)
 708  FORMAT(/,' CALCULATE THE BALANCE WEIGHTS')
      WRITE(10,1610)
 1610 FORMAT(/,' THE ORIG. UNBALANCE RESPONSE  AS A COMPLEX
     1 NUMBER IS:',/)
      WRITE(10,*)XX(1),XX(2),XX(3),XX(4),XX(5),XX(6)
```

```
      WRITE(10,710)
  710 FORMAT(/,' THE RESPONSE FROM 1ST TRIAL WT AS CMPLX
     1 #',/)
      WRITE(10,*)X11,X21,X31,X41,X51,X61
      WRITE(10,712)
  712 FORMAT(/,' THE RESPONSE FROM 2ND TRIAL WT AS CMPLX
     1 #',/)
      WRITE(10,*)X12,X22,X32,X42,X52,X62
      WRITE(10,7130)
 7130 FORMAT(/,' THE RESPONSE FROM 3RD TRIAL WT AS CMPLX
     1 #',/)
      WRITE(10,*)X13,X23,X33,X43,X53,X63
C
C     EXPRESS TRIAL WEIGHT FORCES  AS COMPLEX QUANTITY
C
      T1X=WT1*ET1*COS(ALPHA1)
      T1Y=WT1*ET1*SIN(ALPHA1)
      T1=CMPLX(T1X,T1Y)
      T2X=WT2*ET2*COS(ALPHA2)
      T2Y=WT2*ET2*SIN(ALPHA2)
      T2=CMPLX(T2X,T2Y)
      T3X=WT3*ET3*COS(ALPHA3)
      T3Y=WT3*ET3*SIN(ALPHA3)
      T3=CMPLX(T3X,T3Y)
      WRITE(10,799)
  799 FORMAT(/,' THE TRIAL WEIGHTS AS COMPLEX QUANTIES')
      WRITE(10,*)T1,T2,T3
C
C     CALCULATE INFLUENCE COEFFICIENT MATRIX
C
      ALPH(1,1)=(X11-X10)/T1
      ALPH(2,1)=(X21-X20)/T1
      ALPH(3,1)=(X31-X30)/T1
      ALPH(4,1)=(X41-X40)/T1
      ALPH(5,1)=(X51-X50)/T1
      ALPH(6,1)=(X61-X60)/T1
      ALPH(1,2)=(X12-X10)/T2
      ALPH(2,2)=(X22-X20)/T2
      ALPH(3,2)=(X32-X30)/T2
      ALPH(4,2)=(X42-X40)/T2
      ALPH(5,2)=(X52-X50)/T2
      ALPH(6,2)=(X62-X60)/T2
      ALPH(1,3)=(X13-X10)/T3
      ALPH(2,3)=(X23-X20)/T3
      ALPH(3,3)=(X33-X30)/T3
      ALPH(4,3)=(X43-X40)/T3
      ALPH(5,3)=(X53-X50)/T3
      ALPH(6,3)=(X63-X60)/T3
      WRITE(10,714)
  714 FORMAT(/,' THE MATRIX OF INFLUENCE COEFFS IS',/)
      DO 716 I=1,6
      WRITE(10,*)ALPH(I,1),ALPH(I,2),ALPH(I,3)
```

```
C       WRITE(11,*)ALPH(I,1),ALPH(I,2)
  716   CONTINUE
C
C       FIND THE BALANCE WEIGHTS
C
        DO 2400 II=1,6
        DO 2400 JJ=1,3
        ALCJ(II,JJ)=CONJG(ALPH(II,JJ))
 2400   CONTINUE
        WRITE(10,718)
  718   FORMAT(/,' THE CONJUGATE OF THE INFLU.COEFF. MATRIX
     1  IS',/)
        DO 720 I=1,6
        WRITE(10,*)ALCJ(I,1),ALCJ(I,2),ALCJ(I,3)
  720   CONTINUE
        ALCJT(1,1)=ALCJ(1,1)
        ALCJT(2,1)=ALCJ(1,2)
        ALCJT(3,1)=ALCJ(1,3)
        ALCJT(1,2)=ALCJ(2,1)
        ALCJT(2,2)=ALCJ(2,2)
        ALCJT(3,2)=ALCJ(2,3)
        ALCJT(1,3)=ALCJ(3,1)
        ALCJT(2,3)=ALCJ(3,2)
        ALCJT(3,3)=ALCJ(3,3)
        ALCJT(1,4)=ALCJ(4,1)
        ALCJT(2,4)=ALCJ(4,2)
        ALCJT(3,4)=ALCJ(4,3)
        ALCJT(1,5)=ALCJ(5,1)
        ALCJT(2,5)=ALCJ(5,2)
        ALCJT(3,5)=ALCJ(5,3)
        ALCJT(1,6)=ALCJ(6,1)
        ALCJT(2,6)=ALCJ(6,2)
        ALCJT(3,6)=ALCJ(6,3)
        WRITE(10,722)
  722   FORMAT(/,' THE TRANSPOSE OF THE CONJUGATE IS;',/)
        DO 724 I=1,3
        WRITE(10,*)ALCJT(I,1),ALCJT(I,2),ALCJT(I,3),ALCJT(I,4)
     1  ,ALCJT(I,5),ALCJT(I,6)
  724   CONTINUE
        CALL MCRCR(3,6,ALCJT,3,6,3,ALPH,6,3,3,PROD,3)
        WRITE(10,726)
  726   FORMAT(/,' [ALCJT] * [ALPH] = [PROD]',/)
        DO 7260 I=1,3
        WRITE(10,*)PROD(I,1),PROD(I,2),PROD(I,3)
 7260   CONTINUE
        CALL LINCG(3,PROD,3,PRODI,3)
        CALL MCRCR(3,3,PROD,3,3,3,PRODI,3,3,3,CHECK,3)
        CALL MCRCR(3,3,PRODI,3,3,6,ALCJT,3,3,6,ALPH2,3)
        CALL MCRCR(3,6,ALPH2,3,6,1,XX,6,3,1,CW,3)
        WRITE(10,*) CW(1),CW(2),CW(3)
        WRITE(10,732)
  732   FORMAT(/,' [CW] = -[ALPH2] * [XX0] ',/)
```

```
      CW(1)=-CW(1)
      CW(2)=-CW(2)
      CW(3)=-CW(3)
      WRITE(10,1500)
 1500 FORMAT(/,' THE SET OF CORRECTION WEIGHTS IS:')
      WRITE(10,*)REAL(CW(1)),AIMAG(CW(1))
      WRITE(10,*)REAL(CW(2)),AIMAG(CW(2))
      WRITE(10,*)REAL(CW(3)),AIMAG(CW(3))
      BAL1=(CABS(CW(1))/et1)
      BAL2=(CABS(CW(2))/et2)
      BAL3=(CABS(CW(3))/et3)
      GAMMA1=ATAN2(AIMAG(CW(1)),REAL(CW(1)))
      GAMMA2=ATAN2(AIMAG(CW(2)),REAL(CW(2)))
      GAMMA3=ATAN2(AIMAG(CW(3)),REAL(CW(3)))
      WRITE(10,*)BAL1,GAMMA1
      WRITE(10,*)BAL2,GAMMA2
      WRITE(10,*)BAL3,GAMMA3
C
C     CALCULATE THE RESIDUAL VIBRATION
C
      CALL MCRCR(6,3,ALPH,6,3,1,CW,3,6,1,ACW,6)
      XR(1,1)=XX(1)+ACW(1,1)
      XR(2,1)=XX(2)+ACW(2,1)
      XR(3,1)=XX(3)+ACW(3,1)
      XR(4,1)=XX(4)+ACW(4,1)
      XR(5,1)=XX(5)+ACW(5,1)
      XR(6,1)=XX(6)+ACW(6,1)
      WRITE(10,1550)
 1550 FORMAT(/,' THE CALCULATED RESIDUAL VIBRATION IS:')
      WRITE(10,1555)
 1555 FORMAT(/,'[XR] = XXO + [A]*[CW]',/)
      WRITE(10,*)XR(1,1),XR(2,1),XR(3,1),XR(4,1)
     1 ,XR(5,1),XR(6,1)
C
C     W=W+DW
C     IF(W.GT.50.0)GOTO 15
   43 CONTINUE
   15 CONTINUE
      STOP
      END



C     SUBROUTINE CMPLY
C
      SUBROUTINE CMPLY(A,B,C,N)
      COMPLEX A(N,N),B(N,N),C(N,N)
C     DO 20 I=1,N
C     DO 20 J=1,N
C   20 WRITE(6,30)REAL(A(I,J)),AIMAG(A(I,J))
C   30 FORMAT(2E10.3)
```

```
      DO 40 I=1,N
      DO 40 J=1,N
      C(I,J)=0
      DO 40 K=1,N
   40 C(I,J)=C(I,J)+A(I,K)*B(K,J)
      RETURN
      END
C
C     SUBROUTINE TRANS(A,B,N)
C     B=TRANSPOSE OF A
C
      SUBROUTINE TRANS(A,B,N)
      COMPLEX A(N,N),B(N,N)
      DO 40 I=1,N
      DO 40 J=1,N
      B(I,J)=A(J,I)
   40 CONTINUE
      RETURN
      END
```